

DATABASE SYSTEMS

Valluvar College of Science & Management, Karur.

Department of Computer Science

Database Systems (16scccs4)

Unit I

Introduction: Database-System Applications- Purpose of Database Systems - View of Data -- Database Languages - Relational Databases - Database Design -Data Storage and Querying Transaction Management -Data Mining and Analysis - Database Architecture - Database Users and Administrators - History of Database Systems.

Unit II

Relational Model: Structure of Relational Databases -Database Schema - Keys - Schema Diagrams - Relational Query Languages - Relational OperationsFundamental Relational-Algebra Operations Additional Relational-Algebra Operations- Extended Relational-Algebra Operations - Null Values - Modification of the Database.

Unit III

SQL:Overview of the SQL Query - Language - SQL Data Definition - Basic Structure of SQL Queries - Additional Basic Operations - Set Operations - Null Values Aggregate Functions - Nested Subqueries - Modification of the Database -Join Expressions - Views - Transactions - Integrity Constraints - SQL Data Types and Schemas - Authorization

Unit IV

Relational Languages: The Tuple Relational Calculus - The Domain Relational Calculus Database Design and the E-R Model: Overview of the Design Process - The Entity-Relationship Model - Reduction to Relational Schemas - Entity-Relationship Design Issues - Extended E-R Features - Alternative Notations for Modeling Data - Other Aspects of Database Design

Unit V

Relational Database Design: Features of Good Relational Designs - Atomic Domains and First Normal Form - Decomposition Using Functional Dependencies - Functional-Dependency Theory - Decomposition Using Functional Dependencies - Decomposition Using Multivalued Dependencies-More Normal Forms - Database-Design Process

2 Marks Question with Answer**UNIT 1: INTRODUCTION**

1. Define database management system?

DBMS is Data Base Management System, which consists of a collection of interrelated data and a set of programs to access those data.

2. List any eight applications of DBMS.

- Banking
- Airlines
- Universities
- Credit card transactions
- Tele communication
- Finance
- Sales
- Manufacturing
- Human resources

3. What are the disadvantages of file processing system?

The disadvantages of file processing systems are

- Data redundancy and inconsistency
- Difficulty in accessing data
- Data isolation
- Integrity problems
- Atomicity problems
- Concurrent access anomalies

4. What are the advantages of using a DBMS?

The advantages of using a DBMS are

- Controlling redundancy
- Restricting unauthorized access
- Providing multiple user interfaces
- Enforcing integrity constraints.
- Providing backup and recovery

5. Give the levels of data abstraction?

- Physical level
- logical level
- view level

6. Define instance and schema?

Instance: Collection of data stored in the data base at a particular moment is called an Instance of the database.

Schema: The overall design of the data base is called the data base schema.

7. Define the terms 1) physical schema 2) logical schema.

Physical schema: The physical schema describes the database design at the physical

level, which is the lowest level of abstraction describing how the data are actually stored.
Logical schema: The logical schema describes the database design at the logical level, which describes what data are stored in the database and what relationship exists among the data.

8. What is conceptual schema?

The schemas at the view level are called sub schemas that describe different views of the database.

9. Define data model?

A data model is a collection of conceptual tools for describing data, data relationships, data semantics and consistency constraints.

10. What is storage manager?

A storage manager is a program module that provides the interface between the Low level data stored in a database and the application programs and queries submitted to the system.

11. What are the components of storage manager?

The storage manager components include

- Authorization and integrity manager
- Transaction manager
- File manager
- Buffer manager

12. What is the purpose of storage manager?

The storage manager is responsible for the following

- Interaction with the file manager
- Translation of DML commands in to low level file system commands
- Storing, retrieving and updating data in the database

13. List the data structures implemented by the storage manager.

The storage manager implements the following data structure

- Data files
- Data dictionary
- Indices

14. What is a data dictionary?

A data dictionary is a data structure which stores meta data about the structure of the database ie. the schema of the database.

15. What is an entity relationship model?

The entity relationship model is a collection of basic objects called entities and relationship among those objects. An entity is a thing or object in the real world that is distinguishable from other objects.

16. What are attributes? Give examples.

An entity is represented by a set of attributes. Attributes are descriptive properties possessed by each member of an entity set.

Example: possible attributes of customer entity are customer name, customer id, customer street, customer city.

17. What is relationship? Give examples

A relationship is an association among several entities.

Example: A depositor relationship associates a customer with each account that he/she has.

18. Define the terms Entity set, Relationship set.

Entity set: The set of all entities of the same type is termed as an entity set.

Relationship set: The set of all relationships of the same type is termed as a relationship set.

19. Define single valued and multivalued attributes.

Single valued attributes: attributes with a single value for a particular entity are called single valued attributes.

Multivalued attributes: Attributes with a set of value for a particular entity are called multivalued attributes.

20. What are stored and derived attributes?

Stored attributes: The attributes stored in a data base are called stored attributes.

Derived attributes: The attributes that are derived from the stored attributes are called derived attributes.

21. What are composite attributes?

Composite attributes can be divided into sub parts.

22. Define null values.

In some cases a particular entity may not have an applicable value for an attribute or if we do not know the value of an attribute for a particular entity. In these cases null value is used.

23. Define the terms Entity type, Entity set.

Entity type: An entity type defines a collection of entities that have the same attributes.

Entity set: The set of all entities of the same type is termed as an entity set.

24. What is meant by the degree of relationship set?

The degree of relationship type is the number of participating entity types.

25. Define weak and strong entity sets?

Weak entity set: entity set that do not have key attribute of their own are called weak entity sets. Strong entity set: Entity set that has a primary key is termed a strong entity set.

UNIT 2: RELATIONAL MODEL

1. What does the cardinality ratio specify?
Mapping cardinalities or cardinality ratios express the number of entities to which another entity can be associated. Mapping cardinalities must be one of the following:
 - One to one
 - One to many
 - Many to one
 - Many to many

2. Define the terms DDL, DML
DDL: Data base schema is specified by a set of definitions expressed by a special language called a data definition language.
DML: A data manipulation language is a language that enables users to access or manipulate data as organized by the appropriate data model.

3. Write short notes on relational model
The relational model uses a collection of tables to represent both data and the relationships among those data. The relational model is an example of a record based model.

4. Define tuple and attribute
Attributes: column headers
Tuple: Row

5. Define the term relation.
Relation is a subset of a Cartesian product of list domains.

6. Define tuple variable
Tuple variable is a variable whose domain is the set of all tuples.

7. Define the term Domain.
For each attribute there is a set of permitted values called the *domain* of that attribute.

8. What is a candidate key?
Minimal super keys are called *candidate keys*.

9. What is a primary key?
Primary key is chosen by the database designer as the principal means of identifying an entity in the entity set.

10. What is a super key?
A *super key* is a set of one or more attributes that collectively allows us to identify uniquely an entity in the entity set.

11. Define- relational algebra.
The relational algebra is a procedural query language. It consists of a set of operations that take one or two relation as input and produce a new relation as output.

12. What is a SELECT operation?

The *select* operation selects tuples that satisfy a given predicate. We use the lowercase letter σ to denote selection.

13. What is a PROJECT operation?

The project operation is a unary operation that returns its argument relation with certain attributes left out. Projection is denoted by π .

14. Write short notes on tuple relational calculus.

The tuple relational calculation is a non-procedural query language. It describes the desired information without giving a specific procedure for obtaining that information.

A query or expression can be expressed in tuple relational calculus as $\{t \mid P(t)\}$

which means the set of all tuples 't' such that predicate P is true for 't'.

Notations used:

- $t[A] \rightarrow$ the value of tuple 't' on attribute, A
- $t \in r \rightarrow$ tuple 't' is in relation 'r'
- $\exists \rightarrow$ there exists

15. Definition for 'there exists' (\exists):

$\exists t \in r(Q(t))$

which means there exists a tuple 't' in relation 'r' such that predicate Q(t) is true.

$\forall \rightarrow$ for all

16. Definition for 'for all' (\forall):

$\forall t \in r(Q(t))$

which means Q(t) is true for all tuples 't' in relation 'r'.

$\Rightarrow \rightarrow$ Implication

17. Definition for Implication (\Rightarrow):

$P \Rightarrow Q$ means if P is true then Q must be true.

18. Write short notes on domain relational calculus

The domain relational calculus uses domain variables that take on values from an attribute domain rather than values for entire tuple.

19. Define query language?

A query is a statement requesting the retrieval of information. The portion of DML that involves information retrieval is called a query language.

20. Write short notes on Schema diagram.

A database schema along with primary key and foreign key dependencies can be depicted pictorially by schema diagram. Each relation appears as a box with attributes listed inside it and the relation name above it.

21. What is foreign key?

A relation schema r1 derived from an ER schema may include among its attributes the primary key of another relation schema r2. This attribute is called a *foreign key* from r1 referencing r2.

UNIT 3: SQL

1. What are the parts of SQL language?

The SQL language has several parts: data - definition language Data manipulation language.

2. What are the categories of SQL command?

SQL commands are divided in to the following categories:

- data - definition language
- data manipulation language
- Data Query language
- data control language
- transaction control statements

3. What are the three classes of SQL expression?

SQL expression consists of three clauses:

1. Select
2. From
3. Where

4. Give the general form of SQL query?

Select A₁, A₂ , An **From** R₁, R₂ , R_m **Where** P

5. What is the use of rename operation?

Rename operation is used to rename both relations and a attributes.

It uses the as clause, taking the form:

Old-name **as** new-name

6. Define tuple variable?

Tuple variables are used for comparing two tuples in the same relation. The tuple variables are defined in the **from** clause by way of the **as** clause.

7. List the string operations supported by SQL?

- Pattern matching Operation
- Concatenation
- Extracting character strings
- Converting between uppercase and lower case letters.

8. List the set operations of SQL?

- Union
- Intersect operation
- The except operation

9. What is the use of Union and intersection operation?

Union: The result of this operation includes all tuples that are either in r1 or in r2 or in both r1 and r2. Duplicate tuples are automatically eliminated.

Intersection: The result of this relation includes all tuples that are in both r1 and r2.

10. What are aggregate functions? And list the aggregate functions supported by SQL?

Aggregate functions are functions that take a collection of values as input and return a single value.

Aggregate functions supported by SQL are Average: avg, Minimum: min, Maximum: max Total: sum, Count: count

11. What is the use of group by clause?

Group by clause is used to apply aggregate functions to a set of tuples. The attributes given in the **group by** clause are used to form groups. Tuples with the same value on all attributes in the **group by** clause are placed in one group.

12. What is the use of sub queries?

A sub query is a select-from-where expression that is nested within another query. A common use of sub queries is to perform tests for set membership, make set comparisons, and determine set cardinality.

13. What is view in SQL? How is it defined?

Any relation that is not part of the logical model, but is made visible to a user as a virtual relation is called a view.

We define view in SQL by using the **create view** command. The form of **create view** command is **Create view** v as <query expression>

14. What is the use of with clause in SQL?

The **with** clause provides a way of defining a temporary view whose definition is available only to the query in which the **with** clause occurs.

15. List the table modification commands in SQL?

- Deletion
- Insertion
- Updates
- Update of a view

16. List out the statements associated with a database transaction?

- Commit work
- Rollback work

17. What is transaction?

Transaction is a unit of program execution that accesses and possibly updates various data items.

18. List the SQL domain Types?

SQL supports the following domain types.

1) Char(n) 2) varchar(n) 3) int 4) numeric(p,d) 5) float(n) 6) date.

19. What is the use of integrity constraints?

Integrity constraints ensure that changes made to the database by authorized users do not result in a loss of data consistency. Thus integrity constraints guard against accidental damage to the database.

20. What are domain constraints?

A domain is a set of values that may be assigned to an attribute .all values that appear in a column of a relation must be taken from the same domain.

21. What are referential integrity constraints?

A value that appears in one relation for a given set of attributes also appears for a certain set of attributes in another relation.

22. What is assertion? Mention the forms available.

An assertion is a predicate expressing a condition that we wish the database always to satisfy. Domain integrity constraints. Referential integrity constraints.

23. What does database security refer to?

Database security refers to the protection from unauthorized access and malicious destruction or alteration.

24. List some security violations (or) name any forms of malicious access.

- Unauthorized reading of data
- Unauthorized modification of data
- Unauthorized destruction of data.

25. List the types of authorization.

- Read authorization
- Write authorization
- Update authorization
- Drop authorization

26. List out various user authorization to modify the database schema.

- Index authorization
- Resource authorization
- Alteration authorization
- Drop authorization

27. Name the various privileges in SQL?

- Delete
- Select
- Insert
- update

28. Mention the various user privileges.

All privileges directly granted to the user or role.

All privileges granted to roles that have been granted to the user or role.

29. Give the limitations of SQL authorization.

- The code for checking authorization becomes intermixed with the rest of the application code.
- Implementing authorization through application code rather than specifying it declaratively in SQL makes it hard to ensure the absence of loopholes.

UNIT 4: RELATIONAL LANGUAGES

1. What is meant by nonprocedural query language?

The tuple relational calculus, by contrast, is a nonprocedural query language. It describes the desired information without giving a specific procedure for obtaining that information. A query in the tuple relational calculus is expressed as: $\{t \mid P(t)\}$

2. Write a query to find the ID, name, dept name, salary for instructors whose salary is greater than \$80,000.

$$\{t \mid t \in \text{instructor} \wedge t[\text{salary}] > 80000\}$$

$$\{t \mid \exists s \in \text{instructor} (t[\text{ID}] = s[\text{ID}] \wedge s[\text{salary}] > 80000)\}$$

3. Formal Definition for tuple relation calculus.

A tuple-relational-calculus expression is of the form: $\{t \mid P(t)\}$ where P is a formula. Several tuple variables may appear in a formula. A tuple variable is said to be a free variable unless it is quantified by a \exists or \forall .

Thus, in: $t \in \text{instructor} \wedge \exists s \in \text{department}(t[\text{dept name}] = s[\text{dept name}])$ t is a free variable. Tuple variable s is said to be a bound variable.

4. What is meant by Safety of Expressions?

A tuple-relational-calculus expression may generate an infinite relation.

The expression: $\{t \mid \neg (t \in \text{instructor})\}$

There are infinitely many tuples that are not in instructor. Most of these tuples contain values that do not even appear in the database.

5. What is meant by Expressive Power of Languages?

The tuple relational calculus restricted to safe expressions is equivalent in expressive power to the basic relational algebra (with the operators \cup , $-$, \times , σ , and π , but without the extended relational operations such as generalized projection and aggregation (G)).

6. Define Domain Relational Calculus.

A second form of relational calculus, called domain relational calculus, uses domain variables that take on values from an attributes domain, rather than values for an entire tuple.

7. Formal Definition for domain relational calculus.

An expression in the domain relational calculus is of the form $\{ \langle x_1, x_2, \dots, x_n \rangle \mid P(x_1, x_2, \dots, x_n) \}$ where x_1, x_2, \dots, x_n represent domain variables. P represents a formula composed of atoms, as was the case in the tuple relational calculus.

8. Rules to build up a domain relational calculus.

- An atom is a formula.
- If P1 is a formula, then so are $\neg P1$ and $(P1)$.
- If P1 and P2 are formulae, then so are $P1 \vee P2$, $P1 \wedge P2$, and $P1 \Rightarrow P2$.
- If $P1(x)$ is a formula in x, where x is a free domain variable, then $\exists x (P1(x))$ and $\forall x (P1(x))$

9. Find the instructorID, name, dept name, and salary for instructors whose salary is greater

than \$80,000.

$$\{ \langle i, n, d, s \rangle \mid \langle i, n, d, s \rangle \in \text{instructor} \wedge s > 80000 \}$$

10. Expressive Power of Languages for domain relational calculus.

When the domain relational calculus is restricted to safe expressions, it is equivalent in expressive power to the tuple relational calculus restricted to safe expressions.

The restricted tuple relational calculus is equivalent to the relational algebra; all three of the following are equivalent:

- The basic relational algebra (without the extended relational-algebra operations)
- The tuple relational calculus restricted to safe expressions
- The domain relational calculus restricted to safe expressions

11. Define entity relationship model.

The entity-relationship (E-R) data model was developed to facilitate database design by allowing specification of an enterprise schema that represents the overall logical structure of a database.

12. Define Entity Sets.

An entity is a “thing” or “object” in the real world that is distinguishable from all other objects. For example, each person in a university is an entity. An entity has a set of properties, and the values for some set of properties may uniquely identify an entity.

13. Define Relationship Sets.

A relationship is an association among several entities. For example, we can define a relationship advisor that associates instructor Katz with student Shankar. This relationship specifies that Katz is an advisor to student Shankar. A relationship set is a set of relationships of the same type. Formally, it is a mathematical relation on $n \geq 2$ (possibly nondistinct) entity sets. If E_1, E_2, \dots, E_n are entity sets, then a relationship set R is a subset of $\{ \langle e_1, e_2, \dots, e_n \rangle \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n \}$ where $\langle e_1, e_2, \dots, e_n \rangle$ is a relationship.

14. What is an attribute?

It is a single-valued property of either an entity-type or a relationship-type. For example, a lecture might have attributes: time, date, duration, place, etc. An attribute is represented by an Ellipse.

15. What is meant by Composite attribute?

It is possible to break down composite attribute. For example, a student's full name may be further divided into first name, second name, and last name.

16. What is meant by Derived attribute?

This type of attribute does not include in the physical database. Their values are derived from other attributes present in the database. For example, age should not be stored directly. Instead, it should be derived from the DOB of that employee.

17. What is meant by Multivalued attribute?

Multivalued attributes can have more than one values. For example, a student can have more than one mobile number, email address, etc.

18. Define Cardinality.

Defines the numerical attributes of the relationship between two entities or entity sets.

Different types of cardinal relationships are:

- One-to-One Relationships
- One-to-Many Relationships
- May to One Relationships
- Many-to-Many Relationships

19. List out the ER- Diagram Notations.

ER- Diagram is a visual representation of data that describe how data is related to each other.

- **Rectangles:** This symbol represent entity types
- **Ellipses :** Symbol represent attributes
- **Diamonds:** This symbol represents relationship types
- **Lines:** It links attributes to entity types and entity types with other relationship types
- **Primary key:** attributes are underlined
- **Double Ellipses:** Represent multi-valued attributes

20. Define Unified Modeling Language UML.

The Unified Modeling Language (UML) is a standard developed under the auspices of the Object Management Group (OMG) for creating specifications of various components of a software system.

21. List out the parts of UML.

- Class diagram. A class diagram is similar to an E-R diagram. Later in this section we illustrate a few features of class diagrams and how they relate to E-R diagrams.
- Use case diagram. Use case diagrams show the interaction between users and the system, in particular the steps of tasks that users perform (such as withdrawing money or registering for a course).
- Activity diagram. Activity diagrams depict the flow of tasks between various components of a system.
- Implementation diagram. Implementation diagrams show the system components and their interconnections, both at the software component level and the hardware component level.

22. Define workflow.

The term workflow refers to the combination of data and tasks involved in processes like those of the preceding examples. Workflows interact with the database system as they move among users and users perform their tasks on the workflow.

23. Define Weak Entities.

A weak entity is a type of entity which doesn't have its key attribute. It can be identified uniquely by considering the primary key of another entity. For that, weak entity sets need to have participation.

24. List out the difference between strong entity and weak entity set.

Strong Entity Set	Weak Entity Set
Strong entity set always has a primary key.	It does not have enough attributes to build a primary key.
It is represented by a rectangle symbol.	It is represented by a double rectangle symbol.
It contains a Primary key represented by the underline symbol.	It contains a Partial Key which is represented by a dashed underline symbol.
The member of a strong entity set is called as dominant entity set.	The member of a weak entity set called as a subordinate entity set.
Primary Key is one of its attributes which helps to identify its member.	In a weak entity set, it is a combination of primary key and partial key of the strong entity set.
In the ER diagram the relationship between two strong entity set shown by using a diamond symbol.	The relationship between one strong and a weak entity set shown by using the double diamond symbol.
The connecting line of the strong entity set with the relationship is single.	The line connecting the weak entity set for identifying relationship is double.

UNIT 5: RELATIONAL DATABASE DESIGN

1. What is meant by normalization of data?
 - It is a process of analyzing the given relation schemas based on their Functional Dependencies (FDs) and primary key to achieve the properties
 - Minimizing redundancy
 - Minimizing insertion, deletion and updating anomalies.

2. What is meant by functional dependencies?

Consider a relation schema R and $\alpha \subset R$ and $\beta \subset R$. The functional dependency $\alpha \rightarrow \beta$ holds on relational schema R if in any legal relation $r(R)$, for all pairs of tuples t_1 and t_2 in r such that $t_1[\alpha] = t_2[\alpha]$, and also $t_1[\beta] = t_2[\beta]$.

3. What are the uses of functional dependencies?
 - To test relations to see whether they are legal under a given set of functional dependencies.
 - To specify constraints on the set of legal relations.

4. Define Boyce codd normal form

A relation schema R is in BCNF with respect to a set F of functional dependencies if, for all functional dependencies $\alpha \rightarrow \beta$ in F, where α

5. List the disadvantages of relational database system
 - Repetition of data
 - Inability to represent certain information.

6. What is first normal form?

The domain of attribute must include only atomic (simple, indivisible) values.

7. Explain trivial dependency?

Functional dependency of the form $\alpha \rightarrow \beta$ is trivial if $\beta \subset \alpha$. Trivial functional dependencies are satisfied by all the relations.

8. What are axioms?

Axioms or rules of inference provide a simpler technique for reasoning about functional dependencies.

9. Define canonical cover?

A canonical cover F_c for F is a set of dependencies such that F logically implies all dependencies in F_c and F_c logically implies all dependencies in F. F_c must have the following properties.

10. List the properties of canonical cover.
 - F_c must have the following properties.
 - No functional dependency in F_c contains an extraneous attribute. Each left side of a functional dependency in F_c is unique.

11. What is meant by normalization of data?

It is a process of analysing the given relation schemas based on their Functional Dependencies

(FDs) and primary key to achieve the properties

- Minimizing redundancy
- Minimizing insertion, deletion and updating anomalies

12. List out the desirable properties of decomposition.

- Lossless-join decomposition
- Dependency preservation
- Repetition of information

13. What is 2NF?

A relation schema R is in 2NF if it is in 1NF and every non-prime attribute A in R is fully functionally dependent on primary key.

14. Define instance.

An instance of a relation that satisfies all such real-world constraints is called a legal instance of the relation; a legal instance of a database is one where all the relation instances are legal instances.

15. What does Boyce-Codd Normal Form (BCNF) mean?

Boyce-Codd Normal Form (BCNF) is one of the forms of database normalization. A database table is in BCNF if and only if there are no non-trivial functional dependencies of attributes on anything other than a superset of a candidate key.

16. Define third normal form.

A relation schema R is in third normal form with respect to a set F of functional dependencies if, for all functional dependencies in F of the form $\alpha \rightarrow \beta$, where $\alpha \subseteq R$ and $\beta \subseteq R$, at least one of the following holds:

- $\alpha \rightarrow \beta$ is a trivial functional dependency.
- α is a superkey for R.
- Each attribute A in $\beta - \alpha$ is contained in a candidate key for R.

17. What is meant by Multivalued Dependencies?

Multivalued dependencies, do not rule out the existence of certain tuples. Instead, they require that other tuples of a certain form be present in the relation. For this reason, functional dependencies sometimes are referred to as equality-generating dependencies, and multivalued dependencies are referred to as tuple-generating dependencies.

DATABASE SYSTEMS (16SCCCS4)

5 MARKS QUESTIONS

UNIT -1

1. Explain about Database-System Applications.
2. Discuss about the architecture of database.
3. Describe about View of Data in DBMS.
4. Discuss about DBMS Schema and instance.
5. Explain about database user and interfaces.
6. Discuss about Database Administrator.

UNIT II

1. Describe the structure of relational database.
2. Explain about database schema.
3. Describe about select operation with eg.
4. Describe about projection operation with eg.
5. Explain about union operation with eg.
6. Describe about set difference operation with eg.
7. Describe about Cartesian product operation with eg.
8. Explain about NULL Values with eg.

UNIT III

1. Discuss about the overview of Sql.
2. Explain about basic datatypes in sql.
3. Write in detail about data definition in Sql.
4. Explain about views with eg.
5. Discuss about set operations with eg.
6. Explain about String operations with eg.
7. Describe about Aggregate function with eg.
8. Explain about transaction in sql.
9. Describe about authorization.

UNIT IV

1. Explain database design & ER model.
2. Discuss about the use of entity set in shape.
3. Explain about the extended ER features.
4. Explain about Formal Definition for tuple relational calculus.
5. Describe about Formal Definition for domain relational calculus.
6. Explain about Safety of Expressions for tuple relational calculus.

7. Describe about Safety of Expressions for domain relational calculus.
8. Discuss about Expressive Power of Languages in domain relational calculus.
9. Explain about types attributes with eg.
10. Discuss about mapping cardinalities with eg.
11. Binary versus n -ary Relationship Sets.

UNIT V

1. Discuss about Decomposition Using Functional Dependencies.
2. Explain about First Normal Form.
3. Discuss about Boyce–Codd Normal Form.
4. Explain about Third Normal Form.
5. Describe about Lossless Decomposition
6. Explain about Multivalued Dependencies.
7. Discuss about Fourth Normal Form.
8. Explain about E-R Model and Normalization.

10 MARKS QUESTIONS

UNIT -1

1. Explain about Database-System Applications.
2. Discuss about the architecture of database.
3. Discuss about database languages.
4. Discuss about Database Administrator.
5. Describe about Purpose of Database Systems.

UNIT II

1. Describe the Modification of the Database.
2. Describe about Algebra Operations with eg.
3. Explain about database schema.
4. Explain about NULL Values with eg.
5. Describe about key Constraints.

UNIT III

1. Discuss about the SQL Data Types and Schemas.
2. Explain about Authorization in sql.
3. Write in detail about Modification of the Database in Sql.
4. Explain about views with eg.
5. Discuss about set operations with eg.
6. Explain about String operations with eg.
7. Describe about Aggregate function with eg.

8. Explain about languages in sql.
9. Discuss about join operation in SQL.

UNIT IV

1. Explain about Tuple Relational Calculus.
2. Discuss about Domain Relational Calculus.
3. Explain about the extended ER features.
4. Explain about Entity-Relationship Model

UNIT V

1. Discuss about Decomposition Using Functional Dependencies.
2. Explain about Normal Form.
3. Discuss about Boyce–Codd Normal Form.
4. Explain about Decomposition Using MultivaluedDependencies.
5. Describe about Database-Design Process.