

# Advanced Database Management Systems

Ashok Arora

This reference book can be useful for  
BBA, MBA, B.Com, BMS, M.Com, BCA, MCA  
and many more courses for Various Universities



**NEERAJ**  
**PUBLICATIONS**  
**[www.neerajbooks.com](http://www.neerajbooks.com)**

Published by:



# NEERAJ PUBLICATIONS

*(Publishers of Educational Books)*

Sales Office : 1507, 1st Floor,

Nai Sarak, Delhi-110 006

E-mail: info@neerajbooks.com

Website: www.neerajbooks.com

© **Reserved with the Publishers only.**

Typesetting by: Competent Computers

## Terms & Conditions for Buying E-Book

- The User must Read & Accept the Terms and Conditions (T&C) carefully before clicking on the accept option for Buying the Online Soft Copy of E-books. Under this Particular Facility you may buy only the Online Soft Copy of E-books, no Hard Copy or Printed Copy shall be provided under this facility.
- These E-Books are valid for 365 days online reading only (From the Date of Purchase) and no kind of Downloading, Printing, Copying, etc. are allowed in this facility as these products are just for Online Reading in your Mobile / Tablet / Computers.
- All the online soft copy E-books given in this website shall contain a diffused watermark on nearly every page to protect the material from being pirated / copy / misused, etc.
- This is a Chargeable Facility / Provision to Buy the Online Soft Copy of E-books available online through our Website Which a Subscriber / Buyer may Read Online on his or her Mobile / Tablet / Computer. The E-books content and their answer given in these Soft Copy provides you just the approximate pattern of the actual Answer. However, the actual Content / Study Material / Assignments / Question Papers might somewhat vary in its contents, distribution of marks and their level of difficulty.
- These E-Books are prepared by the author for the help, guidance and reference of the student to get an idea of how he/she can study easily in a short time duration. Content matter & Sample answers given in this E-Book may be Seen as the Guide/Reference Material only. Neither the publisher nor the author or seller will be responsible for any damage or loss due to any mistake, error or discrepancy as we do not claim the Accuracy of these solution / Answers. Any Omission or Error is highly regretted though every care has been taken while preparing these E-Books. Any mistake, error or discrepancy noted may be brought to the publishers notice which shall be taken care of in the next edition. Please consult your Teacher/Tutor or refer to the prescribed & recommended study material of the university / board / institute / Govt. of India Publication or notification if you have any doubts or confusions before you appear in the exam or Prepare your Assignments before submitting to the University/Board/Institute.
- Publisher / Study Badshah / shall remain the custodian of the Contents right / Copy Right of the Content of these reference E-books given / being offered at the website [www.studybadshah.com](http://www.studybadshah.com).
- The User agrees Not to reproduce, duplicate, copy, sell, resell or exploit for any commercial purposes, any portion of these Services / Facilities, use of the Service / Facility, or access to the Service / Facility.
- The Price of these E-books may be Revised / Changed without any Prior Notice.
- The time duration of providing this online reading facility of 365 days may be alter or change by [studybadshah.com](http://studybadshah.com) without any Prior Notice.
- The Right to accept the order or reject the order of any E-books made by any customer is reserved with [www.studybadshah.com](http://www.studybadshah.com) only.
- All material prewritten or custom written is intended for the sole purpose of research and exemplary purposes only. We encourage you to use our material as a research and study aid only. Plagiarism is a crime, and we condone such behaviour. Please use our material responsibly.
- In any Dispute What so ever Maximum Anyone can Claim is the Cost of a particular E-book which he had paid to Study Badshah company / website.
- If In case any Reader/Student has paid for any E-Book and is unable to Access the same at our Website for Online Reading Due to any Technical Error/ Web Admin Issue / Server Blockage at our Website [www.studybadshah.com](http://www.studybadshah.com) then He will be send a New Link for that Particular E-Book to Access the same and if Still the Issue is Not Resolved Because of Technical Error/ Web Admin Issue / Server Blockage at our website then His Amount for that Particular Purchase will be refunded by our website via PayTM.
- All the Terms, Matters & Disputes are Subjected to "Delhi" Jurisdiction Only.

# CONTENTS

<b>S.No.</b>	<b>Page</b>
1. Relational Database Design .....	1
2. Database Implementation and Tools .....	9
3. Advanced SQL .....	16
4. Database System Catalogue .....	23
5. Query Processing and Evaluation .....	29
6. Transaction Management and Recovery .....	37
7. Database Security and Authorisation .....	52
8. Distributed Databases .....	57
9. Object-Oriented Database .....	68
10. Database and XML .....	76
11. Introduction to Data Warehousing .....	86
12. Introduction to Data Mining .....	94
13. Emerging Database Models, Technologies and Applications-I .....	100
14. Emerging Database Models, Technologies and Applications-II .....	107
15. PostgreSQL .....	112
16. Oracle .....	116

# Sample Preview of The Chapter

*Published by:*



**NEERAJ  
PUBLICATIONS**

[www.neerajbooks.com](http://www.neerajbooks.com)

# ADVANCED DATABASE MANAGEMENT SYSTEMS

**DATABASE DESIGN AND IMPLEMENTATION**



## Relational Database Design

### INTRODUCTION

A relational database is designed to minimise data redundancy, improve data consistency and sharing, data independence, etc. Enhanced-relationship (EER) model results from extending the original E-R model. In this chapter, we will learn the various tools for EER modeling. We will mainly discuss about functional dependency, multivalued dependency, normalisation, various normal forms, join dependency, inclusion dependency and temporal data.

### CHAPTER AT A GLANCE

#### FEATURES OF GOOD DATABASE DESIGN

The various features of a good database design are as follows:

- Minimal data redundancy
- Improved data consistency
- Improved data sharing
- Improved data quality
- Improved data accessibility and responsiveness
- Enforcement of standards
- Increased productivity of application development
- Reduced program maintenance
- Data independence

#### ENHANCED ER TOOLS

Enhanced-relationship (EER) model is the model that has resulted from extending the original E-R model with the following new modeling constructs:

1. Supertype/Subtype relationship
2. Inheritance
3. Specialisation
4. Generalisation

**1. Supertype/Subtype Relationship:** The E-R model has been extended to include Supertype/Subtype relationships. A supertype is a generic entity type that has a relationship with one or more subtypes, whereas a subtype is a sub-grouping of those entities in an entity type, which is meaningful to the organisation.

**2. Inheritance:** Inheritance is the property by which subtype entities inherit values of all attributes of the supertype. A subtype is an entity type in its own right. An entity instance of a subtype represents the same entity instance of the supertype. An entity in a subtype must possess not only values for its own attributes, but also values for its attributes as a member of the supertype. The inheritance property makes it unnecessary to include supertype attributes redundantly with the subtypes.

**3. Specialisation:** Specialisation is the top-down process of defining one or more subtypes of the supertype, and forming supertype/subtype relationships. Each subtype is formed based on some

2 / NEERAJ : ADVANCED DATABASE MANAGEMENT SYSTEMS

distinguishing features such as attributes or relationships that are specific to the subtype.

**4. Generalisation:** Generalisation is the bottom-up process of defining a more general entity type from a set of more specialised entity types. Thus, generalisation groups the entity types along with their common attributes, while at the same time it preserves attributes that are specific to each subtype.

**Completeness Constraint**

Completeness constraint means whether an instance of a supertype must also be a member of at least one subtype. It has two possibilities: Total specialisation rule and Partial specialisation rule.

**Disjoint Constraint**

Disjointness constraint means whether an instance of a supertype may simultaneously be a member of two or more subtypes. It also has two possibilities: Disjoint rule and Overlap rule.

**Converting EER Diagram to Table**

EER diagram can be converted into a table by creating a table for each strong and each weak entity set. A table is also created for each binary  $m : n$  relationship set and a binary  $m : 1$  or  $1 : m$  relationship. For generalisation or specialisation hierarchy separate tables can be created for higher level and lower level entities. A separate table is created for composite attributes.

**FUNCTIONAL DEPENDENCY: THEORY AND NORMALISATION**

A functional dependency is a constraint between two attributes or two sets of attributes. It means for any relation  $R$ , attribute  $B$  is functionally dependent on attribute  $A$  if, for every valid instance of  $A$ , that value of  $A$  uniquely determines the value of  $B$ . The functional dependency of  $B$  on  $A$  is represented as follows:

$$A \rightarrow B$$

An attribute may be functionally dependent on two or more attributes. The attribute on the left-hand side of the arrow in a functional dependency is called a determinant. So,  $A$  is a determiner.

**Primary Key, Composite Key and Candidate Key**

Primary key is an attribute or a combination of attributes that uniquely identifies each row in a table. For example, ISBN could be a primary key in the relation 'Book'. Composite key is a primary key that consists of more than one attribute. For example, Account\_Type and ACNO could be a composite key in relation 'Account'.

A candidate key is an attribute, or combination or attributes, that uniquely identifies a row in a relation. For example, Vehicle\_No could be a candidate key in relation 'Vehicle' if all other attributes functionally dependent on Vehicle\_No.

**Normalisation**

Normalisation is the process of decomposing relations with anomalies to produce smaller, well-structured relations. It is based on the analysis of functional dependencies. It can be accomplished in stages, each of which is called a normal form. A normal form is a state of a relation that results from applying simple rules regarding functional dependencies to that relation. The various normal forms are:

1. First normal form
2. Second normal form
3. Third normal form
4. Boyce/Codd normal form
5. Fourth normal form
6. Fifth normal form

**MULTIVALUED DEPENDENCIES AND FOURTH NORMAL FORM**

**Boyce-Codd Normal Form**

A relation is in Boyce-Codd normal form (BCNF) if and only if every determinant in the relation is a candidate key.

**Fourth Normal Form**

When a relation is in BCNF, there are no more any anomalies, which result from functional dependencies. But, some anomalies may result from multi-valued dependencies. A type of dependency that exists when there are at least three attributes (such as  $A$ ,  $B$ , and  $C$ ) in a relation, and for each value of  $A$  there is a well-defined set of values of  $B$  and a well-defined set of values of  $C$ . But the set of values of  $B$  is independent of set  $C$ , and *vice versa*. To remove the multivalued dependency from a relation, the relation is divided into two new relations. Each of these tables contains two attributes that have a multivalued relationship in the original relation. So, a relation is in fourth normal form (4NF) if it is in BCNF and contains no multivalued dependencies.

**JOIN DEPENDENCIES AND 5NF/PJNF**

The fifth normal form deals with a property called Lossless Joins, which occurs very rarely and is difficult to detect in practice. It deals with join-dependencies

and cannot be constructed from several smaller relations. A relation R satisfies join dependency  $*$   $(R_1, R_2, \dots, R_n)$  if and only if R is equal to the join of  $R_1, R_2, \dots, R_n$  where  $R_i$  are subsets of the set of attributes of R. A relation R is in fifth normal form if for all join dependencies, either  $(R_1, R_2, \dots, R_n)$  is a trivial join-dependency or every  $R_i$  is a candidate key for R.

Fifth normal form is also called **Project-Join Normal Form (PJNF)**. It is also defined as follows:

A relation schema R having a set F of functional, multivalued, and join dependencies, is in PJNF (5NF), if for all the join dependencies in the closure of F that are of the form,  $*$  $(R_1, R_2, \dots, R_n)$  where  $R_i$  are subsets of the set of attributes of R and either  $(R_1, R_2, \dots, R_n)$  is a trivial join-dependency or every  $R_i$  is a super key for R.

**Definitions of PJNF from the viewpoint of the decomposition**

**Definition 1:** A join dependency  $*$  $(R_1, R_2, \dots, R_n)$  over a relation R is trivial if it is satisfied by every relation  $r(R)$ .

**Definition 2:** A join dependency  $*$  $(R_1, R_2, \dots, R_n)$  over a relation R applies to a relation scheme R if  $R = R_1, R_2, \dots, R_n$ .

**Definition 3:** If R be a relation scheme having F as the set of Functional Dependencies and Join Dependencies over R, R will be in project-join normal form (PJNF) if for every Join Dependency  $*$  $(R_1, R_2, \dots, R_n)$  which can be derived by F that applies to R, either the Join Dependency is trivial or every  $R_i$  is a super key for R.

**Definition 4:** If R be a relation scheme having F as the set of Functional Dependencies and Join Dependencies over R, R will be in project-join normal form (PJNF) if for every Join Dependency  $*$  $(R_1, R_2, \dots, R_n)$  which can be derived by F that applies to R, is implied by the key Functional Dependencies of R.

**INCLUSION DEPENDENCIES AND TEMPLATE DEPENDENCIES**

Inclusion dependency formalise two important types of interrelation constraints that exist between relations – Referential integrity constraints and Class/subclass relationships.

**Inclusion Dependency**

An inclusion dependency,  $R.X < S.Y$  between two sets of attributes (X of a relation schema R, and Y of a relation schema S) is defined as follows:

If  $r$  and  $s$  are the relation state of R and S respectively at any specific time then  $x(r(R)) = y(s(S))$

**Template Dependencies**

The template dependencies generalises the concepts of Inclusion Dependencies. A template dependency resembles a tableau. It is representation of the statement that a relation is invariant under a certain tableau mapping. It consists of a number of hypothesis rows that define certain variables with a special row at the bottom, which is called the conclusion row.

A relation  $r$  satisfies a template dependency, if and only if, a valuation that successfully maps the hypothesis rows to tuple in a relation  $r$ , finds a map for conclusion row to a tuple in  $r$ .

A template dependency (TD) on a relation scheme R is a pair  $T = (T, w)$  where  $T = \{w_1, w_2, \dots, w_k\}$  is a set of hypothesis rows on R, and  $w$  is a single conclusion row on R. A relation  $r(R)$  satisfies TD T if for every valuation  $p$  of T such that  $p(T) = r$ ,  $p$  can be extended to show that  $p(w) \in r$ .

A template dependency is trivial if every relation over R satisfies it.

**Difference between Template Dependency and Tableau**

A Template Dependency differs from a tableau in the ways that a variable like  $(a, b, c, \dots)$  in the conclusion row need not appear in any of the hypothesis row, and variables may not be necessarily restricted to a single column.

**DOMAIN KEY NORMAL FORM (DKNF)**

DKNF is an attempt to define an ultimate normal form that takes into account all possible types of dependencies and constraints. A set of relations that are in DKNF must be free of anomalies. It is an important normal form as it offers a complete solution to avoid the anomalies.

**MODELING TEMPORAL DATA**

Temporal databases are those database applications that involve some aspect of time. In a temporal database system, we need to define database as a sequence of time based data in chronological order and need to resolve events that happen at the same time. The SQL support for temporal data includes data types such as DATE, TIME, TIMESTAMP, INTERVAL, PERIOD, etc.

**CHECK YOUR PROGRESS**

**Q. 1. What is the use of EER diagram?**

**Ans.** Enhanced-relationship (EER) model extends the original E-R model with the new modeling constructs such as Supertype /Subtype relationship, Inheritance, Specialisation, and Generalisation.

**Q. 2. What are the constraints used in EER diagrams?**

**Ans.** The basic constraints used in EER diagram are completeness and disjoint constraints.

**Completeness Constraint:** Completeness constraint means whether an instance of a supertype must also be a member of at least one subtype. It has two possibilities:

(a) **Total specialisation rule:** It specifies that each entity instance of the supertype must be a member of some subtype in the relationship.

(b) **Partial specialisation rule:** It specifies that an entity instance of the supertype cannot belong to any subtype.

**Disjoint Constraint:** Disjointness constraint means whether an instance of a supertype may simultaneously be a member of two or more subtypes. It also has two possibilities:

(a) **Disjoint rule:** It specifies that if an entity instance of the supertype is a member of one subtype, it cannot simultaneously be a member of any other subtype.

(b) **Overlap rule:** It specifies that an entity instance can simultaneously be a member of two subtypes.

**Q. 3. How is an EER diagram converted into a table?**

**Ans.** EER diagram can be converted into a table by following these rules:

1. A table is created for each strong entity set.
2. A table is created for a weak entity set by including the primary key of the strong entity on which it depends.
3. A table is created for each binary  $m : n$  relationship set having the primary keys of both the participating entities.
4. For a binary  $m : 1$  or  $1 : m$  relationship, the primary key on the  $m$  side should be used as the primary key.
5. For generalisation or specialisation hierarchy separate tables can be created for higher level and lower level entities.

6. A separate table is created for composite attributes.

**Q. 4. What are Multi-valued Dependencies? When we can say that a constraint X is multi-determined?**

**Ans.** A multi-valued dependency is a constraint due to multi-valued attributes. A relation is in Boyce-Codd normal form (BCNF) if and only if every determinant in the relation is a candidate key. When a relation is in BCNF, there are no more any anomalies, which result from functional dependencies. But, some anomalies may result from multi-valued dependencies. A type of dependency that exists when there are atleast three attributes (such as A, B, and C) in a relation, and for each value of A there is a well-defined set of values of B and a well-defined set of values of C. But the set of values of B is independent of set C, and *vice versa*. To remove the multivalued dependency from a relation, the relation is divided into two new relations. Each of these tables contains two attributes that have a multivalued relationship in the original relation. So, a relation is in fourth normal form (4NF) if it is in BCNF and contains no multivalued dependencies.

A relation must have atleast three attributes out of which two should be multi-valued. So, a constraint X is multi-determined if atleast two of its attributes are multi-valued.

**Q. 5 Decompose the following into 4NF.**  
EMP

ENAME	PNAME	DNAME
Dev	X	Sanju
Dev	Y	Sainyam
Dev	X	Sainyam
Dev	Y	Sanju

**Ans.**

**EMP PROJECTS**

ENAME	PNAME
DeV	X
Dev	Y

**EMP DEPENDENTS**

ENAME	DNAME
Dev	Saniu
Dev	Sainyam

**Q. 6. Does the following relation satisfy MVDs with 4NF? Decompose the following relation into 5NF.**