

NEERAJ[®]

INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

M.C.S.-23

**Chapter Wise Reference Book
Including Solved Sample Papers**

By: Neeraj Mishra

Based on _____

I.G.N.O.U.

& Various Central, State & Other Open Universities



**NEERAJ
PUBLICATIONS**

(Publishers of Educational Books)



Retail Sales Office:

1507, 1st Floor, Nai Sarak, Delhi - 6

Mob.: 8510009872, 8510009878

E-mail: info@neerajbooks.com

Website: www.neerajbooks.com

MRP ` 350/-

Published by:



NEERAJ PUBLICATIONS

(Publishers of Educational Books)

Retail Sales Office : 1507, 1st Floor, Nai Sarak, Delhi-110 006

E-mail: info@neerajbooks.com | Website: www.neerajbooks.com

© Copyright Reserved with the Publishers only.

Reprint Edition with Updation of Sample Question Paper Only

Typesetting by: Competent Computers, Printed at: Novelty Printing Press

Disclaimer/T&C

1. For the best & up-to-date study & results, please prefer the recommended textbooks/study material only.
2. This book is just a Guide Book/Reference Book published by NEERAJ PUBLICATIONS based on the suggested syllabus by a particular Board/University.
3. These 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 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 Solutions/Answers. Any Omission or Error is highly regretted though every care has been taken while preparing, printing, composing and proofreading of these Books. As all the Composing, Printing, Publishing and Proof Reading, etc., are done by Human only and chances of Human Error could not be denied. Any mistake, error or discrepancy noted may be brought to the publishers notice which shall be taken care of in the next edition and thereafter as a good gesture by our company he/she would be provided the rectified Book free of cost. 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 regarding any information, data, concept, results, etc. before you appear in the exam or Prepare your Assignments before submitting to the University/Board/Institute.
4. In case of any dispute whatsoever the maximum anybody can claim against NEERAJ PUBLICATIONS is just for the price of the Book.
5. The number of questions in NEERAJ study materials are indicative of general scope and design of the question paper.
6. Any type of ONLINE Sale/Resale of "NEERAJ BOOKS" published by "NEERAJ PUBLICATIONS" on Websites, Web Portals, Online Shopping Sites, like Amazon, Flipkart, Ebay, Snapdeal, etc., is strictly not permitted without prior written permission from NEERAJ PUBLICATIONS. Any such online sale activity by an Individual, Company, Dealer, Bookseller, Book Trader or Distributor will be termed as ILLEGAL SALE of NEERAJ BOOKS and will invite legal action against the offenders.
7. The User agrees Not to reproduce, duplicate, copy, sell, resell or exploit for any commercial purposes, any portion of these Books without the written permission of the publisher. This book or part thereof cannot be translated or reproduced in any form (except for review or criticism) without the written permission of the publishers.
8. 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.
9. All matters, terms & disputes are subject to Delhi Jurisdiction only.

Get books by Post & Pay Cash on Delivery :

If you want to Buy NEERAJ BOOKS by post then please order your complete requirement at our Website www.neerajbooks.com where you can select your Required NEERAJ BOOKS after seeing the Details of the Course, Subject, Printed Price & the Cover-pages (Title) of NEERAJ BOOKS.

While placing your Order at our Website www.neerajbooks.com You may also avail the "Special Discount Schemes" being offered at our Official website www.neerajbooks.com.

No need to pay in advance as you may pay "Cash on Delivery" (All The Payment including the Price of the Book & the Postal Charges, etc.) are to be Paid to the Delivery Person at the time when You take the Delivery of the Books & they shall Pass the Value of the Goods to us. We usually dispatch the books Nearly within 3-4 days after we receive your order and it takes Nearly 4-5 days in the postal service to reach your Destination (In total it take nearly 8-9 days).

CONTENTS

INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

Question Bank – (Previous Year Solved Question Papers)

<i>Question Paper—Exam Held in February-2021 (Solved)</i>	1-13
<i>Question Paper—June, 2019 (Solved)</i>	1-7
<i>Question Paper—December, 2018 (Solved)</i>	1-8
<i>Question Paper—June, 2018 (Solved)</i>	1-9
<i>Question Paper—December, 2017 (Solved)</i>	1-7
<i>Question Paper—June, 2017 (Solved)</i>	1-11
<i>Question Paper—December, 2016 (Solved)</i>	1-9
<i>Question Paper—June, 2016 (Solved)</i>	1-8
<i>Question Paper—December, 2015 (Solved)</i>	1-9
<i>Question Paper—June, 2015 (Solved)</i>	1-10
<i>Question Paper—June, 2014 (Solved)</i>	1-10
<i>Question Paper—December, 2013 (Solved)</i>	1-9
<i>Question Paper—December, 2012 (Solved)</i>	1-8
<i>Question Paper—December, 2011 (Solved)</i>	1-11

<i>S.No.</i>	<i>Chapterwise Reference Book</i>	<i>Page</i>
1.	Basic Concepts	1
2.	Relational and E-R Models	12
3.	Database Integrity and Normalization	33
4.	File Organization in DBMS	42
5.	Structured Query Language (SQL)	60
6.	Transactions and Concurrency Management	78

<i>S.No.</i>	<i>Chapter</i>	<i>Page</i>
7.	Database Recovery and Security	91
8.	Distributed and Client Server Databases	98
	● System/Software Life Cycle (Appendix - A)	112
	● Drawing E-R Diagram for a Given Case	122
	● Publishing Export House Stock Maintenance: A Case Study	129
		■ ■

**Sample Preview
of the
Solved
Sample Question
Papers**

Published by:



**NEERAJ
PUBLICATIONS**

www.neerajbooks.com

QUESTION PAPER

*Exam Held in
February – 2021*

(Solved)

INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS (M.C.S.-23)

Time: 3 Hours]

[Maximum Marks: 100

Note : Question No. 1 is compulsory. Attempt the three questions from the rest.

Q. 1. (a) Determine the output when the following operations are applied on Relation R_1 , R_2 and R_3 given below:

$R_1: A$	B	$R_2: X$	Y	$R_3: A$	→ empty
a_1	b_1	a_1	b_1		
a_2	b_2	a_2	b_2		
a_3	b_3	a_3	b_3		
a_4	b_4	a_4	b_4		

(i) Union ($R_1 \cup R_2$)

(ii) Intersection ($R_1 \cap R_2$)

(iii) Difference ($R_1 - R_2$)

(iv) Cartesian Cross Section ($R_1 \times R_2$)

(v) Division ($R_1 \div R_2$)

Ans. (i) Union ($R_1 \cup R_2$)

A	B
A_1	B_1
A_2	B_2
A_3	B_3
A_4	B_4
A_7	B_7

(ii) Intersection ($R_1 \cap R_2$)

A	B
A_1	B_1
A_2	B_2
A_4	B_4

(iii) Difference ($R_1 - R_2$)

A	B
A_3	B_3

(iv) Cartesian Cross Section ($R_1 \times R_2$)

Ans. If R_1 and R_2 are two functions or relations, then the result of $R_3 = R_1 \times R_2$ is the combination of tuples that are in R_1 and R_2 . The product is commutative and associative. Degree (R_3) = Degree of (R_1) + Degree (R_2). In other words, R_3 will have tuples such that $R_3 = \{t_1 \parallel t_2 \mid R_1 \ni t_1 \wedge R_2 \ni t_2\}$.

(v) Division ($R_1 \div R_2$)

Ans. $R_1 :$

A	B
a_1	b_1
a_2	b_2
a_3	b_3
a_4	b_4

(b) How to weak entities differ from strong entities? Discuss with the help of an example. Briefly discuss the role of keys in identification of the entity type, i.e. weak or strong.

Ans. A weak entity is dependent on a strong entity to ensure its existence. Unlike a strong entity, a weak entity does not have any primary key. It instead has a partial discriminator key. A weak entity is represented by a double rectangle. The relation between one strong and one weak entity is represented by a double diamond. Difference Between Strong and Weak Entity:



Talking about the Strong and Weak entity, we must know what does an entity means. An entity is a unique object in the real world. It is described as the set of attributes. The collection of the same type of entities together forms the entity set.

We have Loan as our weak entity, and as we said above for each loan there must be at least one

borrower. You can observe in the loan entity set, no customer has borrowed a car loan and hence, it has totally vanished from loan entity set. For the presence of car loan in loan entity set, it must have been borrowed by a customer. In this way, the weak Loan entity is dependent on the strong customer entity.

Loan_name	Loan_date	Amount
Home	20/11/2015	20000
Education	5/10/2015	10000
Home	20/11/2015	20000

Loan Entity Set

The second thing, we know is a weak entity does not have a primary key. So here Loan_name, the

partial key of the weak entity and Cust_ID primary key of customer entity makes primary key of loan entity.

In the Loan entity set, we have two exactly same entities i.e. a **Home loan on date 20/11/2015 with amount 20000**. Now how to identify who had borrowed them this can be done with the help of primary key of the weak entity (Loan_name + Cust_ID). So, it will be determined that one home loan is borrowed by Customer 101 John and other by Customer 103 Ruby. This is how the composed primary key of weak entity identify each entity in weak entity set.

Basis for Comparison	Strong Entity	Weak Entity
Basic	The strong entity has a primary key.	The weak entity has a partial discriminator key.
Depends	The strong entity is independent of any other entity in a schema.	Weak entity depends on the strong entity for its existence.
Denoted	Strong entity is denoted by a single rectangle.	Weak entity is denoted with the double rectangle.
Relation	The relation between two strong entities is denoted by a single diamond simply called relationship.	The relationship between a weak and a strong entity is denoted by identifying relationship denoted with double diamond.
Participation	Strong entity may or may not have total participation in the relationship.	Weak entity always has total participation in the identifying relationship shown by double line.

(c) What do you understand by functional dependency in a Relational DBMS? For the relation given below, check whether the given functional dependencies hold or not. Give proper justification.

J	K	L
x	1	2
x	1	3
y	1	4
y	1	3
z	2	5
p	4	7

- (i) $J \rightarrow K$
- (ii) $K \rightarrow J$

(iii) $J, K \rightarrow L$

(iv) $K, L \rightarrow J$

Ans. A Functional Dependency (FD) is a relationship between two attributes, typically between the PK and other non-key attributes within a table. For any relation R, attribute Y is functionally dependent on attribute X (usually the PK), if for every valid instance of X, that value of X uniquely determines the value of Y. This relationship is indicated by the representation below :

$X \rightarrow Y$

Sample Preview of The Chapter

Published by:



**NEERAJ
PUBLICATIONS**

www.neerajbooks.com

INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

Basic Concepts



INTRODUCTION

The era in which we are living is called “Information era”. In the age of this “Information era” there is need of instant availability of information for decision making. To make the information instantly available to us, the Databases along with computer, and telecommunications technology play an important role. As computers hence become a part and parcel part of our everyday life so does the databases. The databases and database systems have become an essential part of our everyday life. For example, when we deposit or withdraw money in/from a bank reserve tickets for train or airplane, get or return books from/to a library, directly or indirectly we have some kinds of interactions with databases. In this case, we interact “customer databases” maintained by banks, “passenger databases” maintained by Railways and Airlines, and “library databases” maintained by libraries. An analogy of a database is a “Telephone Directory” where subscribers names and their relevant numbers are listed in alphabetic order. This “Telephone Directory” can be stored in a computer in the form a database and called “Telephone Directory Database” and then can be arranged by subscribers name for the ease and fast access.

Databases such as “Telephone Directory Database” were only capable to store textual and numeric data *e.g.*, subscribers names and their numbers respectively. These types of databases are called **Traditional databases**. But with the advent of new technologies such as multi-media, you can store pictures, video clips,

sound messages along with text and numbers. Such types of databases are called **Multi-media**.

DATABASES

And last, but not least, there is another kind of databases called **Real Time Databases** that can control industrial and manufacturing process.

In this chapter we will discuss

- What a Database is?
- Need for a DBMS;
- Structure of DBMS;
- Three-tier architecture of DBMS; and
- Data Models.

WHAT IS A DATABASE?

A database is a collection of relevant information stored so that it could be available to many users for different purposes. The content of a database called “data” is collected from all the different sources within and outside organization. A computerized database enables the user to get the required information instantly. A database can handle inventory, accounting and filing and use the information in its files to prepare summaries, estimates and reports.

Databases have already a null defined market for specific information for highly selected group of users on almost all the subjects. Such a database is MEDLINE that provides medical information for doctors. Likewise WEST LAW is a computer based information service that caters to the requirements of lawyers.

2 / NEERAJ : INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

WHAT IS DBMS?

The key to success of a database is the manner in which information within it is arranged. The management of data in a database system is done by means of a general purpose software package called **Database Management System (DBMS)**. Thus a **Database Management System (DBMS)** is a software that can be used to set up and monitor a database and can manage the updation and retrieval of data that has been stored in it.

ORACLE, SQL Server, dBASE, Foxbase, Foxpro, MS-Access, Ingers etc. are some commercially available DBMS.

Most DBMSs provide following facility to users:

- Create ion, deletion and duplication (copying) of databases.
- Isertion, deletion, modification and preventing duplication of data in a database.
- Arrangement of data in a particular order (either in ascending or descending order) in a database.
- Recovery of data from system failures.
- Retrieval of data from a database.
- Report generation based on data stored in a database.
- Sharing of data stored in a database among users.

NEED OF A DBMS

Any organization be it a Government Department, Bank, University, Hospital or Bank, require a large amount of data to collect, store and process them as and when required in future. This is because they have a need of instant availability of information. This is why there is a need of DBMS.

FILE ORIENTED APPROACH VS. DATABASE ORIENTED APPROACH

The convention approach for data process is to store data locally and develop a program (or many programs) for each type of application. In the past, different users had programs that handled their own independent stored data. Consequently, there were more than one data files with the same data used by different applications. This is the factor that resulted in **data redundancy**. An example will clarify it better. Suppose, a payroll application program is used by Accounts department and an Employee record keeping system is used by the Personnel department in an organization.

Both applications have their master files containing common data such as employee names, addresses, designation, department, salary, employee code, etc. Although the data required by the two applications are same, but data will have to be stored in two different places in a computer, because each application separately stores and maintains data. Is in not a case of data redundancy/duplicacy? Indeed it is.

An organization needs flow of information across the applications and this requires sharing of data, which is significantly lacking in the traditional/convention “File oriented approach” towards data processing. One major limitations of this approach is that the programs become dependent on the files and the files become dependent on the programs.

The problem pointed out with the File-oriented approach to data processing can be solved using **Database approach**. A **database** is a persistent collection of logically related data. Databases are self-describing in nature. A database contains not only the data but also the complete definition of database structure and constraints, which are stored in a system catalog. A DBMS manages this data and allows data sharing and integration of data of an organization within a single database. A DBMS facilitates user to create databases, modify the structure of a database and its data value stored in it, insert data in a database, retrieve data from a database inforce the data integrity within a database, etc. In brief, database, approach to data processing reduces data redundancy, makes the data sharable and independent, provides data integrity and provides improved security and backup and recovery, which will be discussed under the subheading “Advantages of Database Approach”.

Limitations of File Oriented Approach

The file based system and file oriented approach to data processing has some limitation or disadvantage, which are as follows:

- **Data redundancy/duplication:** Due to decentralization of data, the file oriented system leads to uncontrolled duplication of data, which results in a wastage of a lot of storage space. It also costs time and money to enter data more than once.
- **Separation and isolation of data and programs:** Data is stored in separate files and each data file requires separate processing program. It becomes complex when the data is required to be retrieved from more than two files as a large amount of data has to be search. Each department needs to keep its files updated separately.
- **Program/Data Dependency:** In the file oriented approach if a data field is to be added to a master data file, all such programs that access the master file would have to be modified to allow for this new field which would have been added to the master record. In this way, programs become dependent on the data files and data files become dependent on programs.

- **File formats incompatibility:** Since the structure of data files are embedded (defined) in application programs, the structure is dependent on application programming languages. The file structure created with one language differs to other. For example, the structure of a file created with COBOL differs from one created with PASCAL and so they are incompatible to each other. This incompatibility makes them difficult to process combinedly. To convert them into some common format there will be need of software to be developed, which is a time consuming and expensive job.
- **Supports predefined queries and reports:** In file-oriented system queries and reports are defined by the application programmer during the development of application program. This is the fact, that in any business organization the type and the requirement number of queries and reports increases as the time passes by. Again any new query or report needed by organization have to be developed by the application program; otherwise no new query or report of the data could be generated.

Advantages of Database Approach

The database approach to data processing has many advantage over file-based on file-oriented approach, which are as follows:

- **Reduction of redundancies:** The main reason of data redundancy in file oriented system to data processing is that each group of users maintains its own files. Maintaining separate copies of similar data, results in not only wastage of storage space but may also results in data inconsistencies. The files that represents same day may become inconsistent as some may be updated whereas others may not be. In database approach data can be stored centrally at a single storage location with controlled redundancy under DBMS, which saves space, thus eliminates redundancy and does not permit inconsistency.
- **Data Sharing:** A DBMS allows any number of users or application programs to share a database under its control. A database belonging to an organization can be shared by all authorized users according to the permissions granted to them by the DBA (Database Administrator).
- **Data Independence:** In file-oriented approach data files depend on the programs and programs depend on data file. This is why modification in either, requires modification in another one.

Database Management Systems database structure and its data from application programs; however application program manipulates data stored in the database. This is called **Data Independence**, where details of data are not exposed. DBMS provides an abstract view of database and hide its details, because it isolates data descriptions from data.

- **Improved Integrity:** Data integrity refers to validity and consistency of data. Data integrity means that data should be accurate and consistent. This is achieved by providing some validation checks or constraints on the database. There are some integrity rules almost provided by all the DBMS that the database is not allowed to violate. Constraints may be applied to data items (fields) within a record and/or on records too for validation of data at the entry time. For example, the basic salary of an employee must be between 5000 and 20000. While entering the data for basic salary in the database, say Employee database, the database must check against this rule (constraint).
- **Efficient data access:** A DBMS is capable to provide services to end users, where they can retrieve the from a database immediately. For this DBMS utilizes techniques to store and retrieve data efficiently at least for unseen queries.
- **Multiple user interface:** A DBMS provides various level of interfaces to different levels of users to work with databases. A DBMS such as Visual Foxpro (6.0) provides:
 - query language such as SQL as an interface for casual users.
 - programming language such as Xbase as an interface for application programmer.
 - Menu and wizards as an interface for novice users.Besides these interfaces, a DBMS should also provide natural language interfaces for stand alone users, which are still not available in standard form.
- **Representation of complex relationship among data:** Since a database may have data that needs to be interrelated with each other in many ways, a DBMS must facilitate the user to establish the relationships among databases and retrieve and update the related data easily and efficiently. For example a “student” database and a “student-course” database may need to be related, where a student may have opted many courses. This is the case of “one-to-many” relationship.

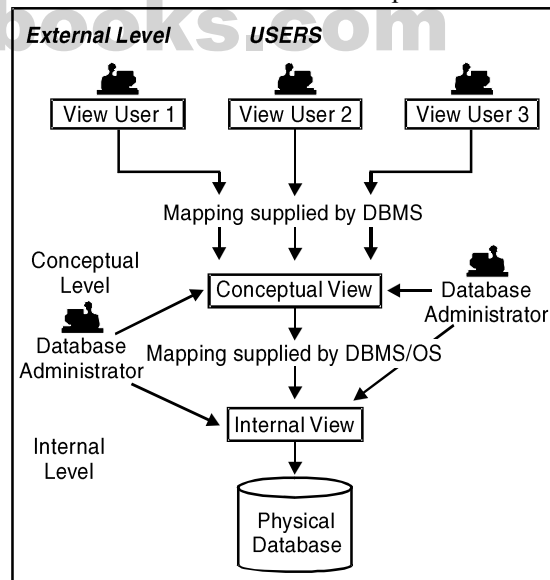
4 / NEERAJ : INTRODUCTION TO DATABASE MANAGEMENT SYSTEMS

- **Security of Database:** Data is not only a vital resource of any organization, but it is confidential too. In a shared system where multiple users access the data, data is subjected to security. All data/information may be accessed by all users; but with restricted permissions. For example, some users may view, delete, modify data records whereas some can only view data records and even some users can not view all the data-fields stored in a database. For examples some users may not be able to see the salary of employees in the “Empolyee”database, whereas the users of Accounts department can. These permissions to the database are granted by the DBA (Database Administrator) to the users. In this way unauthorized access to the database is prohibited by a DBMS. For security and authorization the DBA assigns usernames and passwords to the authorized users and grants permissions/privileges as will for accessing a database. DBA implements it using security and authorization subsystem of the DBMS.
- **Improved Backup and Recovery:** A file-based system usually fails to provide measures to protect data from system failure and it left this responsibility solely on user. So the user of a file-based system is required to take the backup of the database periodically. But, a DBMS provides facilities for recovering the hardware and software failures. Each and every DBMS has a Backup and Recovery subsystem, which is responsible for recover the database from system failure. In case of system failure this subsystem restores the database to a state in which it was before the failure event occurred.
- **Support concurrency control:** A DBMS supports multiple transactions to occur simultaneously.
- **Logical or Three-Level Architecure of DBMS :** There are two different views to look at the architecturs of a DBMS : the logical DBMS architecture and physical DBMS architecture. The logical architecture deals with the way data is stored and presented to the users. It describes how data in a database is perceived by users. It is not concerned with the subject that how data is handled and processed by the DBMS. The physical architeturcs concerned with the software components that make up a DBMS. The method of data storage on the underlying

file system is not revealed by the DBMS to the users and it facilitates users to manipulate data without worrying about where it (data) is stored and how it is actually stoned. This results in the database having different levels of abstraction. A majority of DBMS's are based on the ANSI/SPARC generalised DBMS architecture. The generalized architecture of a database system is called ANSI/SPARC model. This model divides a DBMS into three levels of abstraction.

- ◆ Internal level or physical level
- ◆ Conceptual level
- ◆ External level or view level. The figures below shows the logical architecture of a typical DBMS as per ANSI/SPARC model

- **The Extenral Level or View Subheading Level :** The external or view level is the highest level of abstraction of database. It is the level of the architecture of a DBMS closest to the users. This is the level which is concerned with the way, which data are viewed/observed by the users. They can be either an application program or an end user. There can be many external views of a database as any number of external **schema** can be defined and they can overlap each other. It consists of the definition of logical records and relationships in the external view. It also contains the methods for deriving the objects such as entities, attributes and relationships in the external view from the conceptual view.



The three-level (logical) architecture for a DBMS as per ANSI/SPARC model