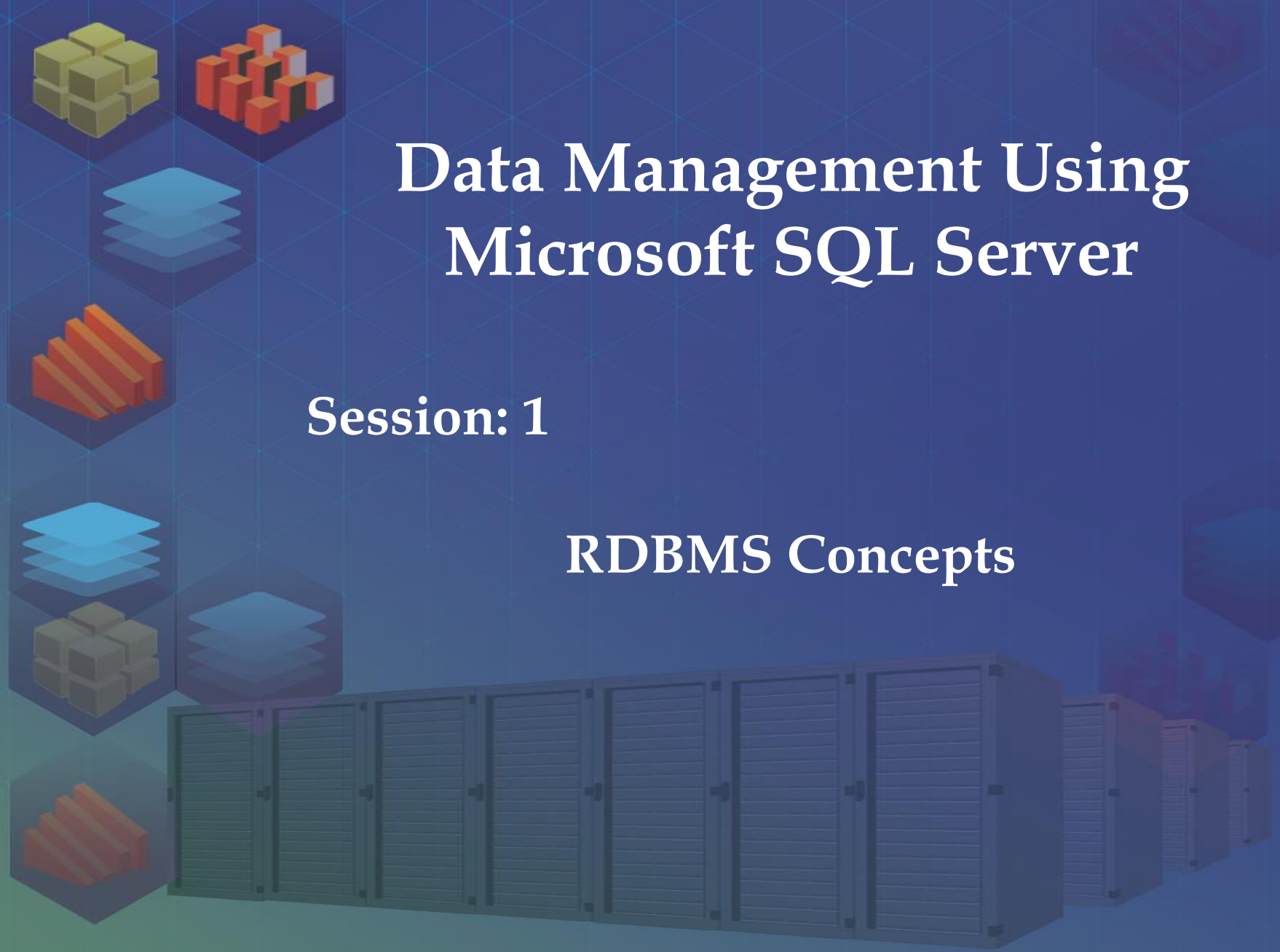


Data Management Using Microsoft SQL Server

Session: 1

RDBMS Concepts

Objectives

- Explain the concept of data and database
- Describe the approaches to data management
- Define a Database Management System (DBMS) and list its benefits
- Explain the different database models
- Define and explain RDBMS
- Describe entities and tables and list the characteristics of tables
- List the differences between a DBMS and an RDBMS

Introduction

- Organizations often maintain large amounts of data, which are generated as a result of day-to-day operations.
- A database:
  - is an organized form of such data.
  - may consist of one or more related data items called records.
  - is a data collection to which different questions can be asked.
- For example,
  - 'What are the phone numbers and addresses of the five nearest post offices?' or
  - 'Do we have any books in our library that deal with health food?'

Data and Database

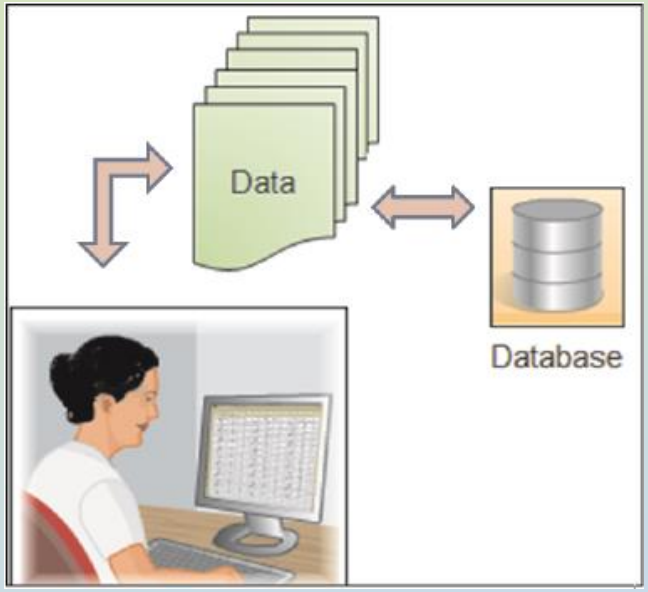
When this data is gathered and analyzed, it yields information. Intelligent interpretation of data yields information.

Information helps to foresee and plan events.

A database is an organized collection of data such that its contents can be easily accessed, managed, and updated.

A **phone book** is a database consisting of names, addresses, and telephone numbers.

- Following figure illustrates the concept of a database:

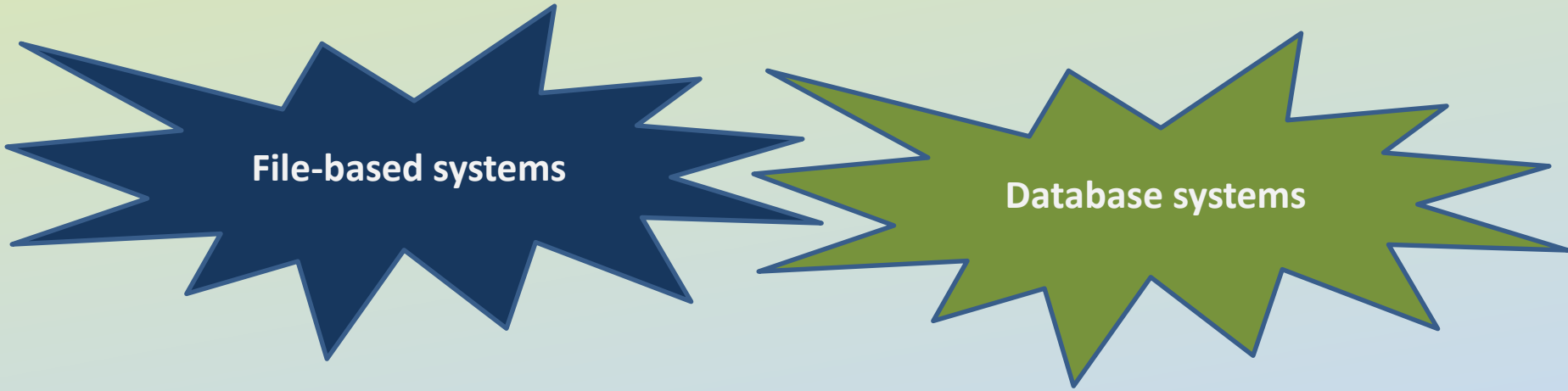


Data Management

Data management deals with managing large amount of information, which involves:

- the storage of information
- the provision of mechanisms for the manipulation of information
- providing safety of information stored under various circumstances

The two different approaches of managing data are as follows:



File-based Systems

In a file-based systems data is stored in discrete files and a collection of such files is stored on a computer.

Files of archived data were called tables because they looked like tables used in traditional file keeping.

Rows in the table were called records and columns were called fields. An example of the file-based system is illustrated in the following table:

First Name	Last Name	Address	Phone
Eric	David	ericd@eff.org	213-456-0987
Selena	Sol	selena@eff.org	987-765-4321
Jordan	Lim	nadroj@otherdomain.com	222-3456-123

Disadvantages of File-based Systems

- Data redundancy and inconsistency
- Unanticipated queries
- Data isolation
- Concurrent access anomalies
- Security problems
- Integrity problems

Database Systems

Database Systems evolved in the late 1960s to address common issues in applications handling large volumes of data, which are also data intensive.

Databases are used to store data in an efficient and organized manner. A database allows quick and easy management of data.

At any point of time, data can be retrieved from the database, added, and searched based on some criteria in these databases.

Data storage can be achieved even using simple manual files.

Data stored in this form is not permanent. Records in such manual files can only be maintained for a few months or few years.

Advantages of Database Systems

- The amount of redundancy in the stored data can be reduced
- No more inconsistencies in data
- The stored data can be shared
- Standards can be set and followed
- Data Integrity can be maintained
- Security of data can be implemented

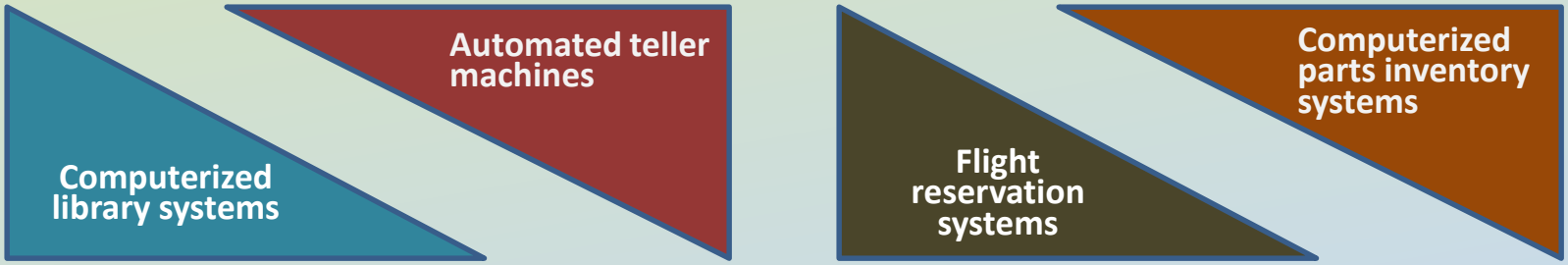
Database Management System (DBMS) 1-2

A DBMS is a collection of related records and a set of programs that access and manipulate these records and enables the user to enter, store, and manage data.

In a centralized database system, the database is stored in the central location which everybody can have access from their machine.

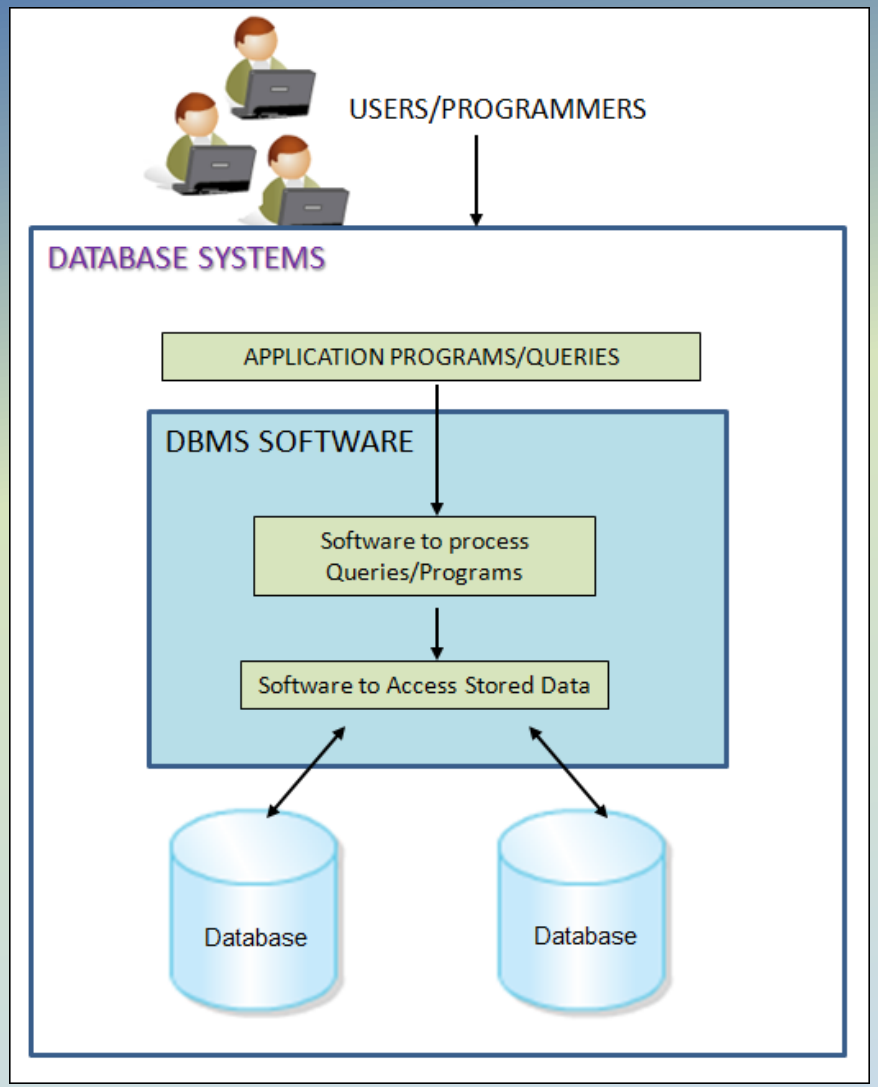
A database is a collection of interrelated data, and a DBMS is a set of programs used to add or modify this data.

Examples of database applications include:



Database Management System (DBMS) 2-2

- Following figure illustrates a database system:



Benefits of DBMS 1-2

A DBMS is responsible for processing data and converting it into information.

A database for this purpose has to be manipulated, which includes querying the database to retrieve specific data, updating the database, and finally, generating reports.

These reports are the source of information, which is, processed data.

A DBMS is also responsible for data security and integrity.

Benefits of DBMS 2-2

- Data storage
- Data definition
- Data manipulation
- Data security and integrity
- Data recovery and concurrency
- Performance optimization
- Multi-user access control
- Database access languages and Application Programming Interfaces (APIs)

Database Models

Databases can be differentiated based on functions and model of the data.

A data model describes a container for storing data, and the process of storing and retrieving data from that container.

The analysis and design of data models has been the basis of the evolution of databases.

Each model has evolved from the previous one. The commonly used Database Models are as follows:

- Flat-file Data Model
- Hierarchical Data Model
- Network Data Model
- Relational Data Model

Flat-file Data Model

In this model, the database consists of only one table or file.

This model is used for simple databases - for example, to store the roll numbers, names, subjects, and marks of a group of students.

This model cannot handle very complex data. It can cause redundancy when data is repeated more than once.

- Following table depicts the structure of a flat file database:

Roll Number	First Name	Last Name	Subject	Marks
45	Jones	Bill	Maths	84
45	Jones	Bill	Science	75
50	Mary	Mathew	Science	80

Hierarchical Data Model 1-3

In this model, different records are inter-related through hierarchical or tree-like structures.

In this model, relationships are thought of in terms of children and parents.

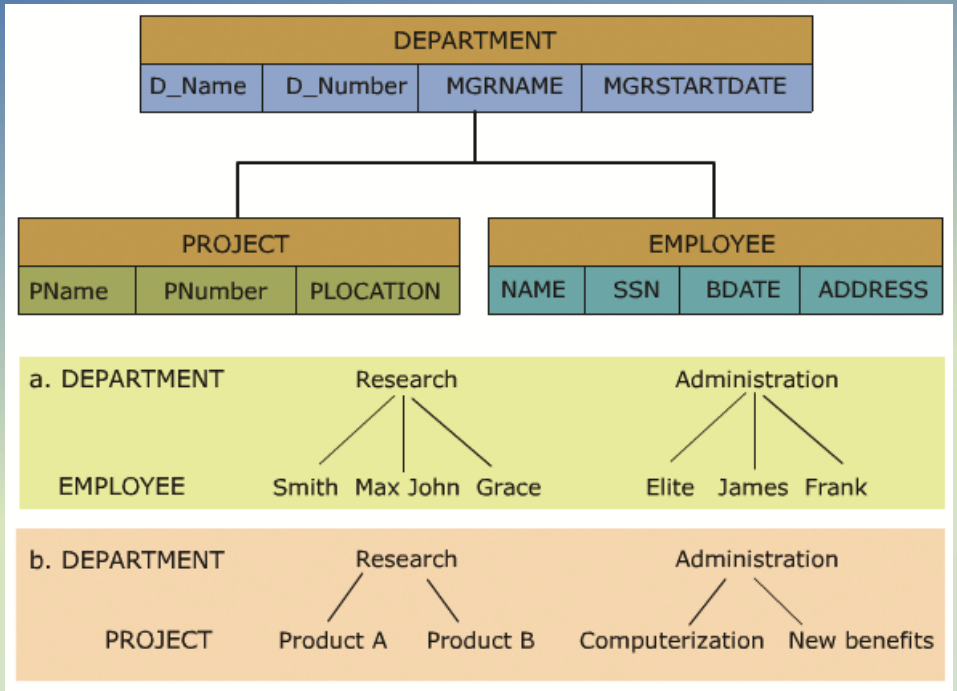
A parent record can have several children, but a child can have only one parent.

To find data stored in this model, the user needs to know the structure of the tree.

Windows Registry is an example of a hierarchical database storing configuration settings and options on Microsoft Windows operating systems.

Hierarchical Data Model 2-3

- Following figure illustrates an example of a hierarchical representation:



- Within the hierarchical model, Department is perceived as the parent of the segment.
- The tables, Project and Employee, are children.
- A path that traces the parent segments beginning from the left, defines the tree.
- This ordered sequencing of segments tracing the hierarchical structure is called the hierarchical path.

Hierarchical Data Model 3-3

- The advantages of a hierarchical model are as follows:

Data is held in a common database so data sharing becomes easier, and security is provided and enforced by a DBMS.

Data independence is provided by a DBMS, which reduces the effort and costs in maintaining the program.

This model is very efficient when a database contains a large volume of data.

- For example, a bank's customer account system fits the hierarchical model well because each customer's account is subject to a number of transactions.

Network Data Model 1-4

This model is similar to the Hierarchical Data Model. It is actually a subset of the network model.

In the network model, data is stored in sets, instead of the hierarchical tree format. This solves the problem of data redundancy.

The set theory of the network model does not use a single-parent tree hierarchy. It allows a child to have more than one parent. Thus, the records are physically linked through linked-lists.

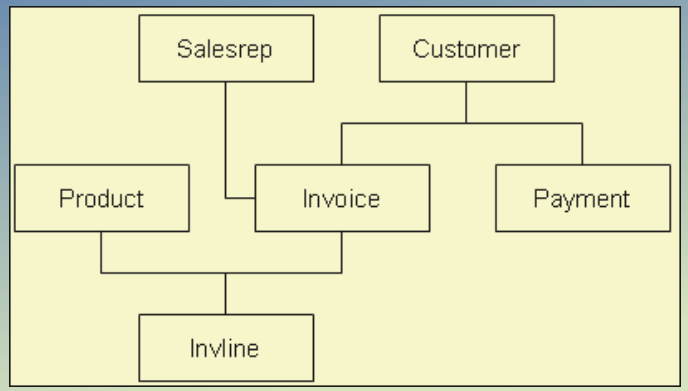
For every database, a definition of the database name, record type for each record, and the components that make up those records is stored. This is called its network schema.

A portion of the database as seen by the application's programs that actually produce the desired information from the data contained in the database is called sub-schema.

It allows application programs to access the required data from the database. Raima Database Manager (RDM) Server by Raima Inc. is an example of a Network DBMS.

Network Data Model 2-4

- The network model shown in the following figure illustrates a series of one-to-many relationships:



- A sales representative may write many **Invoice** tickets, but each Invoice is written by a single Sales representative (**Salesrep**).
- A **Customer** might make purchases on different occasions.
- A **Customer** may have many **Invoice** tickets, but each Invoice belongs only to a single customer.
- An **Invoice** ticket may have many Invoice lines (**Invline**), but each **Invline** is found on a single **Invoice** ticket.
- A **Product** may appear in several different **Invline**, but each **Invline** contains only a single **Product**.

Network Data Model 3-4

- The components of the language used with network models are as follows:

Data Definition Language (DDL)

- Used to create and remove databases and database objects. It enables the database administrator to define the schema components.

Sub-schema DDL

- Enables the database administrator to define the database components.

Data Manipulation Language (DML)

- Used to insert, retrieve, and modify database information.

Data Control Language (DCL)

- Used to administer permissions on the databases and database objects.

Network Data Model 4-4

- The advantages of such a structure are specified as follows:

  - Relationships are easier to implement in the network database model than in the hierarchical model.
  - This model enforces database integrity.
  - This model achieves sufficient data independence.

- The disadvantages are specified as follows:

  - The databases in this model are difficult to design.
  - The programmer has to be familiar with the internal structures to access the database.
  - The model provides a navigational data access environment.

- This model is difficult to implement and maintain.
- Computer programmers, rather than end users, utilize this model.

Relational Data Model 1-5

As the information needs grew and more sophisticated databases and applications were required, database design, management, and use became too cumbersome.

This led to the development of what came to be called the Relational Model database.

The term 'Relation' is derived from the set theory of mathematics. In the Relational Model, unlike the Hierarchical and Network models, there are no physical links.

All data is maintained in the form of tables consisting of rows and columns. Data in two tables is related through common columns and not physical links.

Operators are provided for operating on rows in tables. This model represents the database as a collection of relations.

Relational Data Model 2-5

A row is called a tuple, a column, an attribute, and the table is called a relation.

The list of values applicable to a particular field is called domain.

Several attributes can belong to the same domain.

The number of attributes of a relation is called degree of the relation.

The number of tuples determines the cardinality of the relation.

Relational Data Model 3-5

- In order to understand the relational model, consider the following **Students** and **Marks** tables:

Roll Number	Student Name
1	Sam Reiner
2	John Parkinson
3	Jenny Smith
4	Lisa Hayes
5	Penny Walker
6	Peter Jordan
7	Joe Wong

Students Table

Roll Number	Marks Obtained
1	34
2	87
3	45
4	90
5	36
6	65
7	89

Marks Table

- The **Students** table displays the **Roll Number** and the **Student Name**, and the **Marks** table displays the **Roll Number** and **Marks** obtained by the students.
- To locate students with marks above 40:
  - First, locate the roll numbers of those who have scored above 50 from the **Marks** table.
  - Second, their names have to be located in the **Students** table by matching the roll number.

Relational Data Model 4-5

- The result is displayed as shown in the following table:

Roll Number	Student Name	Marks Obtained
2	John	87
4	Lisa	90
6	Peter	65
7	Joe	89

- It was possible to get this information because of two facts:

First, there is a column common to both the tables - `Roll Number`.

Second, based on this column, the records from the two different tables could be matched and the required information could be obtained.

- In a relational model, data is stored in tables.
- A table in a database has a unique name that identifies its contents.
- Each table can be defined as an intersection of rows and columns.

Relational Data Model 5-5

Advantages of the relational model

- Gives the programmer time to concentrate on the logical view of the database rather than being bothered about the physical view.
- Provides querying flexibility and hence the popularity of the relational databases.
- Easy to handle model to the extent that even untrained people find it easy to generate handy reports and queries, without giving much thought to the need to design a proper database.

Disadvantages of the relational model

- Hides all the complexities of the system and hence it tends to be slower than the other database systems.

Relational Database Management System (RDBMS) 1-3

Relational Model is an attempt to simplify database structures.

Represents all data in the database as simple row-column tables of data values.

An RDBMS is a software program that helps to create, maintain, and manipulate a relational database.

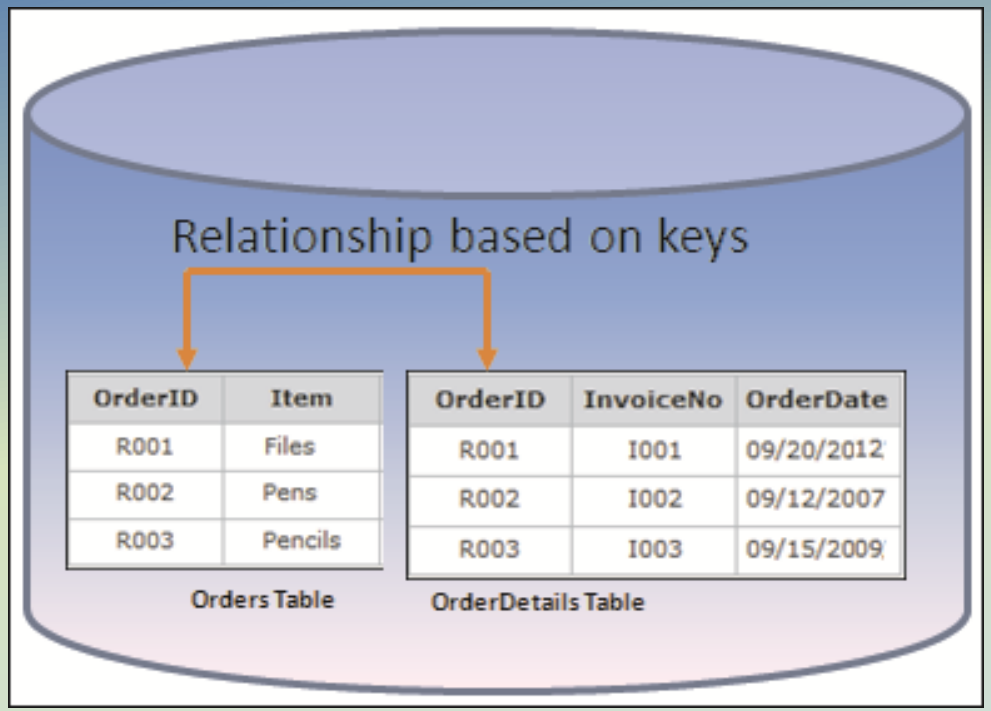
A relational database is a database divided into logical units called tables, where tables are related to one another within the database.

Tables are related in a relational database, allowing adequate data to be retrieved in a single query (although the desired data may exist in more than one table).

By having common keys, or fields, among relational database tables, data from multiple tables can be joined to form one large resultset.

Relational Database Management System (RDBMS) 2-3

- Following figure shows two tables related to one another through a common key (data value) in a relational database:



- Thus, a relational database is a database structured on the relational model.
- Basic characteristic of a relational model is that in a relational model, data is stored in relations.

Relational Database Management System (RDBMS) 3-3

- Following are the **Capitals** and **Currency** tables showing a list of countries and their capitals, and the countries and the local currencies used by them respectively:

Country	Capital
Greece	Athens
Italy	Rome
USA	Washington
China	Beijing
Japan	Tokyo
Australia	Sydney
France	Paris

Capitals Table

Country	Currency
Greece	Drachma
Italy	Lira
USA	Dollar
China	Renminbi (Yuan)
Japan	Yen
Australia	Australian Dollar
France	Francs

Currency Table

- Both the tables have a common column, that is, the **Country** column.
- Now, to display the information about the currency used in Rome, first find the name of the country to which Rome belongs from table **Capitals**.
- Next, that country should be looked up in table **Currency** to find out the currency.
- It is possible to get this information because it is possible to establish a relation between the two tables through a common column called **Country**.

Terms Related to RDBMS 1-3

- There are certain terms that are mostly used in an RDBMS. These are described as follows:

Data is presented as a collection of relations.

Each relation is depicted as a table.

Columns are attributes.

Rows ('tuples') represent entities.

Every table has a set of attributes that are taken together as a 'key' (technically, a 'superkey'), which uniquely identifies each entity.

Terms Related to RDBMS 2-3

- Consider the scenario of a company maintaining customer and order information for products being sold and customer-order details for a specific month, say, August.
- The following tables are used to illustrate this scenario:

Cust_No	Cust_Name	Phone No
002	David Gordon	0231-5466356
003	Prince Fernandes	0221-5762382
003	Charles Yale	0321-8734723
002	Ryan Ford	0241-2343444
005	Bruce Smith	0241-8472198

Customer

Item_No	Description	Price
HW1	Power Supply	4000
HW2	Keyboard	2000
HW3	Mouse	800
SW1	Office Suite	15000
SW2	Payroll Software	8000

Items

Ord_No	Item_No	Qty
101	HW3	50
101	SW1	150
102	HW2	10
103	HW3	50
104	HW2	25
104	HW3	100
105	SW1	100

Order_Details

Ord_No	Ord_Date	Cust_No
101	02-08-12	002
102	11-08-12	003
103	21-08-12	003
104	28-08-12	002
105	30-08-12	005

Order_August

Terms Related to RDBMS 3-3

- Following table lists the terms related to tables:

Term	Meaning	Example from the Scenario
Relation	A table	Order_August, Order_Details, Customer and Items
Tuple	A row or a record in a relation	A row from Customer relation is a Customer tuple
Attribute	A field or a column in a relation	Ord_Date, Item_No, Cust_Name, and so on
Cardinality of a relation	The number of tuples in a relation	Cardinality of Order_Details relation is 7
Degree of a relation	The number of attributes in a relation	Degree of Customer relation is 3
Domain of an attribute	The set of all values that can be taken by the attribute	Domain of Qty in Order_Details is the set of all values which can represent quantity of an ordered item
Primary Key of a relation	An attribute or a combination of attributes that uniquely defines each tuple in a relation	Primary Key of Customer relation is Cust_No Ord_No and Item_No combination forms the primary key of Order_Details
Foreign Key	An attribute or a combination of attributes in one relation R1 that indicates the relationship of R1 with another relation R2. The foreign key attributes in R1 must contain values matching with those of the values in R2	Cust_No in Order_August relation is a foreign key creating reference from Order_August to Customer. This is required to indicate the relationship between orders in Order_August and Customer

RDBMS Users 1-2

- Many persons are involved in the design, use, and maintenance of a large database with a few hundred users.

Database Administrator (DBA)

- Collects the information that will be stored in the database
- Responsible for authorizing access to the database
- Coordinating and monitoring its use
- Acquiring software and hardware resources as needed
- Accountable for problems such as breach of security or poor system response time

Database Designer

- Responsible for identifying the data to be stored in the database
- Choosing appropriate structures to represent and store this data
- Communicate with all prospective database users, in order to understand their requirements
- To come up with a design that meets the requirements

RDBMS Users 2-2

System Analysts and Application Programmers

- Determine the requirements of end users
- Develop specifications for pre-determined transactions that meet these requirements
- Implement these specifications as programs
- Test, debug, document, and maintain these pre-determined transactions
- Design, development, and operation of the DBMS software and system environment

DBMS Designers and Implementers

- Design and implement the DBMS modules and interfaces as a software package.

End User

- The end user invokes an application to interact with the system, or writes a query for easy retrieval, modification, or deletion of data.

Entity

An entity is a person, place, thing, object, event, or even a concept, which can be distinctly identified.

For example, the entities in a university are students, faculty members, and courses.

Each entity has certain characteristics known as attributes.

For example, the student entity might include attributes like student number, name, and grade. Each attribute should be named appropriately.

A grouping of related entities becomes an entity set. Each entity set is given a name. The name of the entity set reflects the contents.

Thus, the attributes of all the students of the university will be stored in an entity set called `Student`.

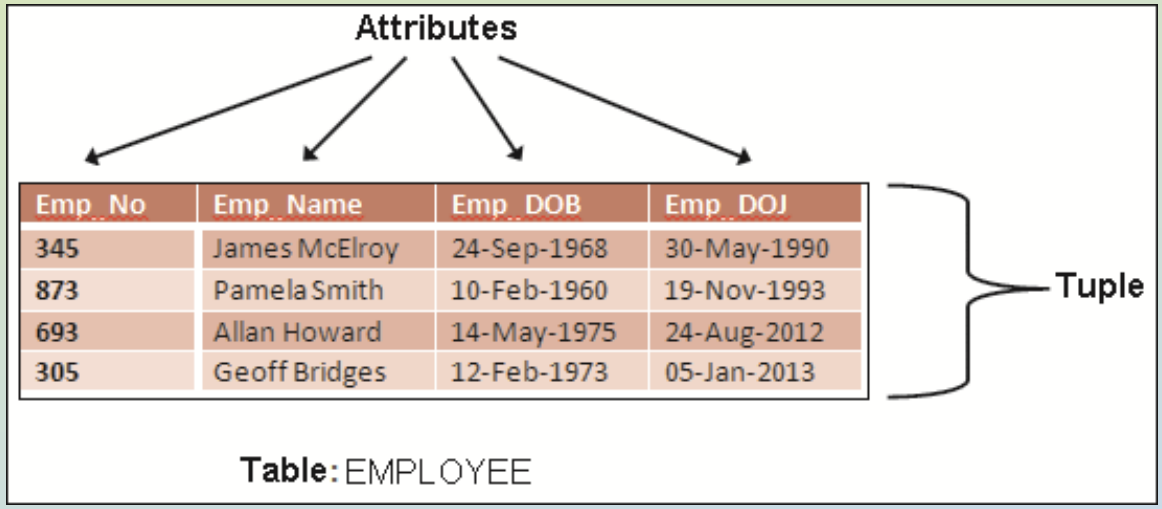
Tables and their Characteristics 1-2

The access and manipulation of data is facilitated by the creation of data relationships based on a construct known as a table.

A table contains a group of related entities that is an entity set. The terms entity set and table are often used interchangeably.

A table is also called a relation. The rows are known as tuples. The columns are known as attributes.

- Following figure highlights the characteristics of a table:



Tables and their Characteristics 2-2

- The characteristics of a table are as follows:

- A two-dimensional structure composed of rows and columns is perceived as a table.
- Each tuple represents a single entity within the entity set.
- Each column has a distinct name.
- Each row/column intersection represents a single data value.
- Each table must have a key known as primary key that uniquely identifies each row.
- All values in a column must conform to the same data format.
- Each column has a specific range of values known as the attribute domain.
- Each row carries information describing one entity occurrence.
- The order of the rows and columns is immaterial in a DBMS.

Differences between a DBMS and an RDBMS

DBMS	RDBMS
It does not need to have data in tabular structure nor does it enforce tabular relationships between data items.	In an RDBMS, tabular structure is a must and table relationships are enforced by the system. These relationships enable the user to apply and manage business rules with minimal coding.
Small amount of data can be stored and retrieved.	An RDBMS can store and retrieve large amount of data.
A DBMS is less secure than an RDBMS.	An RDBMS is more secure than a DBMS.
It is a single user system.	It is a multiuser system.
Most DBMSs do not support client/server architecture.	It supports client/server architecture.
Here, entities are given more importance and there is no relation established among these entities.	Here, a relation is given more importance. Thus, the tables in an RDBMS are dependent and the user can establish various integrity constraints on these tables so that the ultimate data used by the user remains correct.

Summary

- A database is a collection of related data stored in the form of a table.
- A data model describes a container for storing data and the process of storing and retrieving data from that container.
- A DBMS is a collection of programs that enables the user to store, modify, and extract information from a database.
- A Relational Database Management System (RDBMS) is a suite of software programs for creating, maintaining, modifying, and manipulating a relational database.
- A relational database is divided into logical units called tables. These logical units are interrelated to each other within the database.
- The main components of an RDBMS are entities and tables.
- In an RDBMS, a relation is given more importance, whereas, in case of a DBMS, entities are given more importance and there is no relation established among these entities.