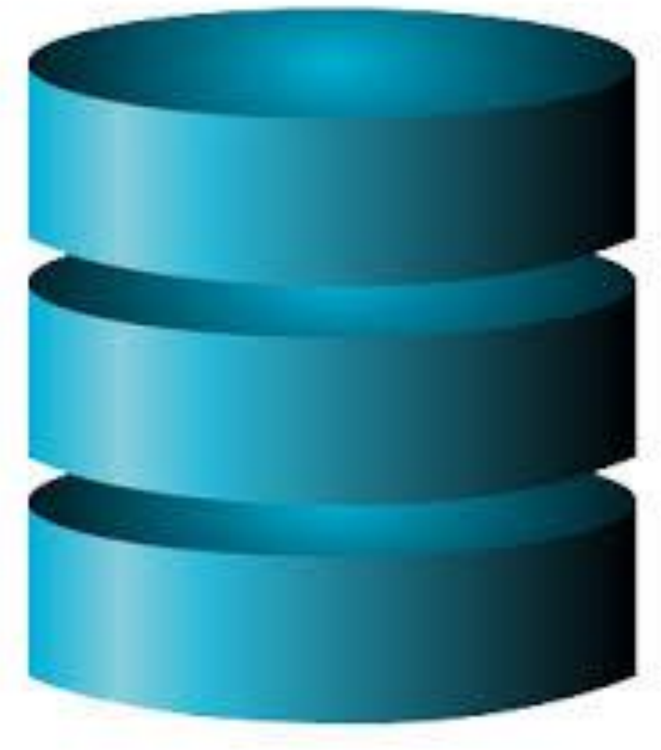
brighten your future....

The Institute of Commerce & Vocational Studies

CHAPTER – 9

DATABASE OPERATIONS

7. Introduction



A database is an organized collection of data, generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal design and modeling techniques.

Advantages of Database

- Reduced data redundancy
- Reduced updating errors and increased consistency
- Greater data integrity and independence from applications programs
- Improved data access to users through use of host and query languages
- Improved data security
- Reduced data entry, storage, and retrieval costs
- Facilitated development of new applications program

8.1 Three levels of database architecture

Following are the three levels of database architecture,

- 1. Physical Level
- 2. Conceptual Level
- 3. External Level

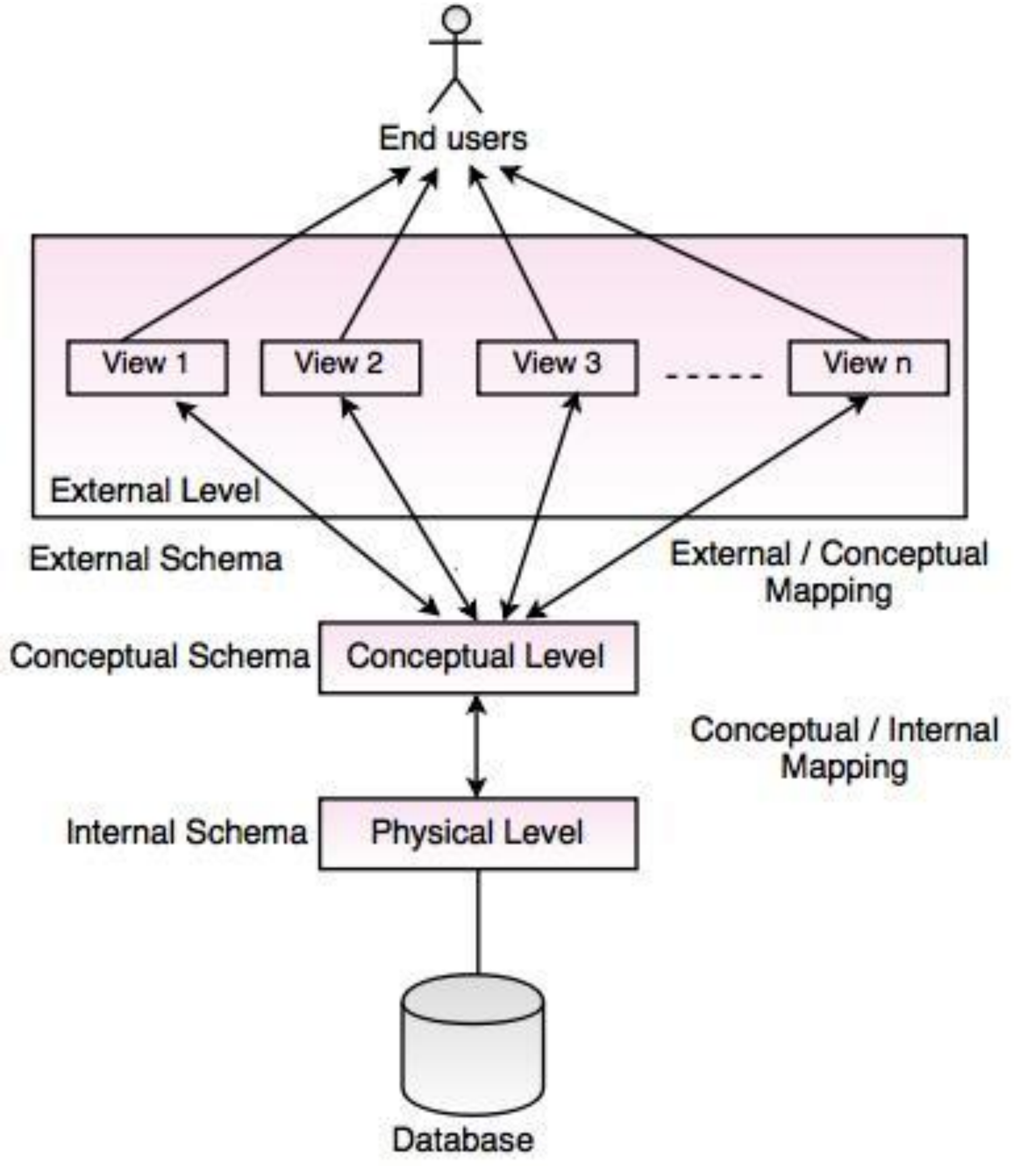


Fig. Three Level Architechture of DBMS

Mapping is the process of transforming request response between various database levels of architecture.

Mapping is not good for small database, because it takes more time.

In External / Conceptual mapping, DBMS transforms a request on an external schema against the conceptual schema.

In Conceptual / Internal mapping, it is necessary to transform the request from the conceptual to internal levels.

Physical Level

Physical level describes the physical storage structure of data in database.

It is also known as Internal Level.

This level is very close to physical storage of data.

At lowest level, it is stored in the form of bits with the physical addresses on the secondary storage device.

At highest level, it can be viewed in the form of files.

The internal schema defines the various stored data types. It uses a physical data model.

Conceptual Level

Conceptual level describes the structure of the whole database for a group of users.

It is also called as the data model.

Conceptual schema is a representation of the entire content of the database.

These schema contains all the information to build relevant external records.

It hides the internal details of physical storage.

External Level

External level is related to the data which is viewed by individual end users.

This level includes a no. of user views or external schemas.

This level is closest to the user.

External view describes the segment of the database that is required for a particular user group and hides the rest of the database from that user group.

Database Models

Database Model

A database model shows the logical structure of a database, including the relationships and constraints that determine how data can be stored and accessed. Individual database models are designed based on the rules and concepts of whichever broader data model the designers adopt. Most data models can be represented by an accompanying database diagram.

Conceptual Data Model

A conceptual data model is a summary-level data model that is most often used on strategic data projects. It typically describes an entire enterprise. Due to its highly abstract nature, it may be referred to as a conceptual model.

Common characteristics of a conceptual data model:

Enterprise-wide coverage of the business concepts. Think Customer, Product, Store, Location, Asset.

Designed and developed primarily for a business audience Contains around 20-50 entities (or concepts) with no or extremely limited number of attributes described. Sometimes architects try to limit it to printing on one page.

Contains relationships between entities, but may or may not include cardinality and nullability.

Entities will have definitions. Designed and developed to be independent of DBMS, data storage locations or technologies. In fact, it would address digital and non-digital concepts. This means it would model paper records and artifacts as well as database artifacts.

Logical Data Model

A logical data model is a fully-attributed data model that is independent of DBMS, technology, data storage or organizational constraints. It typically describes data requirements from the business point of view. While common data modeling techniques use a relational model notation, there is no requirement that resulting data implementations must be created using relational technologies.

Physical Data Model

A physical data model is a fully-attributed data model that is dependent upon a specific version of a data persistence technology. The target implementation technology may be a relational DBMS, an XML document, a NoSQL data storage component, a spreadsheet or any other data implementation option.

7.3 RDBMS

RDBMS

RDBMS stands for Relational Database Management Systems..

All modern database management systems like SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL and Microsoft Access are based on RDBMS.

It is called Relational Data Base Management System (RDBMS) because it is based on relational model introduced by E.F. Codd.

7.4 Relational Data Integrity

Entity Integrity

Entity integrity defines each row to be unique within its table. No two rows can be the same.

To achieve this, a primary key can be defined. The primary key field contains a unique identifier – no two rows can contain the same unique identifier.

Referential Integrity

Referential integrity is concerned with relationships. When two or more tables have a relationship, we have to ensure that the foreign key value matches the primary key value at all times. We don't want to have a situation where a foreign key value has no matching primary key value in the primary table. This would result in an orphaned record.

So referential integrity will prevent users from: Adding records to a related table if there is no associated record in the primary table. Changing values in a primary table that result in orphaned records in a related table. Deleting records from a primary table if there are matching related records.

7.6 Codd’s Rule

Codd's Rule

Dr Edgar F. Codd, after his extensive research on the Relational Model of database systems, came up with twelve rules of his own, which according to him, a database must obey in order to be regarded as a true relational database.

These rules can be applied on any database system that manages stored data using only its relational capabilities. This is a foundation rule, which acts as a base for all the other rules.

Rule 1: Information Rule

The data stored in a database, may it be user data or metadata, must be a value of some table cell. Everything in a database must be stored in a table format.

Rule 2: Guaranteed Access Rule

Every single data element (value) is guaranteed to be accessible logically with a combination of table-name, primary-key (row value), and attribute-name (column value). No other means, such as pointers, can be used to access data.

Rule 3: Systematic Treatment of NULL Values

The NULL values in a database must be given a systematic and uniform treatment. This is a very important rule because a NULL can be interpreted as one the following − data is missing, data is not known, or data is not applicable.

Rule 4: Active Online Catalog

The structure description of the entire database must be stored in an online catalog, known as data dictionary, which can be accessed by authorized users. Users can use the same query language to access the catalog which they use to access the database itself.

Rule 5: Comprehensive Data Sub-Language Rule

A database can only be accessed using a language having linear syntax that supports data definition, data manipulation, and transaction management operations. This language can be used directly or by means of some application. If the database allows access to data without any help of this language, then it is considered as a violation.

Rule 6: View Updating Rule

All the views of a database, which can theoretically be updated, must also be updatable by the system.

Rule 7: High-Level Insert, Update, and Delete Rule

A database must support high-level insertion, updation, and deletion. This must not be limited to a single row, that is, it must also support union, intersection and minus operations to yield sets of data records.

Rule 8: Physical Data Independence

The data stored in a database must be independent of the applications that access the database. Any change in the physical structure of a database must not have any impact on how the data is being accessed by external applications.

Rule 9: Logical Data Independence

The logical data in a database must be independent of its user’s view (application). Any change in logical data must not affect the applications using it. For example, if two tables are merged or one is split into two different tables, there should be no impact or change on the user application. This is one of the most difficult rule to apply.

Rule 10: Integrity Independence

A database must be independent of the application that uses it. All its integrity constraints can be independently modified without the need of any change in the application. This rule makes a database independent of the front-end application and its interface.

Rule 11: Distribution Independence

The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only. This rule has been regarded as the foundation of distributed database systems.

Rule 12: Non-Subversion Rule

If a system has an interface that provides access to low-level records, then the interface must not be able to subvert the system and bypass security and integrity constraints.

7.7 Structured Query Language

SQL

SQL stands for Structured Query Language

SQL lets you access and manipulate databases

SQL became a standard of the American National Standards Institute (ANSI) in 1986,

and of the International Organization for Standardization (ISO) in 1987

SQL can do

- SQL can execute queries against a database
- SQL can retrieve data from a database
- SQL can insert records in a database
- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database
- SQL can set permissions on tables, procedures, and views

Characteristics of SQL

- SQL is easy to learn.
- SQL is used to access data from relational database management systems.
- SQL can execute queries against the database.
- SQL is used to describe the data.
- SQL is used to define the data in the database and manipulate it when needed.
- SQL is used to create and drop the database and table.
- SQL is used to create a view, stored procedure, function in a database.
- SQL allows users to set permissions on tables, procedures, and views.

Advantages of SQL

There are the following advantages of SQL:

High speed : Using the SQL queries, the user can quickly and efficiently retrieve a large amount of records from a database.

No coding needed : In the standard SQL, it is very easy to manage the database system. It doesn't require a substantial amount of code to manage the database system.

Well defined standards : Long established are used by the SQL databases that are being used by ISO and ANSI.

Portability : SQL can be used in laptop, PCs, server and even some mobile phones.

Interactive language : SQL is a domain language used to communicate with the database. It is also used to receive answers to the complex questions in seconds.

Multiple data view : Using the SQL language, the users can make different views of the database structure.

Query

- A query is a request for data results, and for action on data. You can use a query to answer a simple question, to perform calculations, to combine data from different tables, or even to add, change, or delete table data.

- As tables grow in size they can have hundreds of thousands of records, which makes it impossible for the user to pick out specific records from that table.

- With a query you can apply a filter to the table's data, so that you only get the information that you want.

- Queries that you use to retrieve data from a table or to make calculations are called select queries.

- Queries that add, change, or delete data are called action queries.
- You can also use a query to supply data for a form or report.
- In a well-designed database, the data that you want to present by using a form or report is often located in several different tables.
- The tricky part of queries is that you must understand how to construct one before you can actually use them.

SUMMARY

In this Chapter you learned

- ❑ Database
- ❑ Codd’s Rule
- ❑ Relationship
- ❑ SQL
- ❑ Access overview
- ❑ Manipulation access 2010

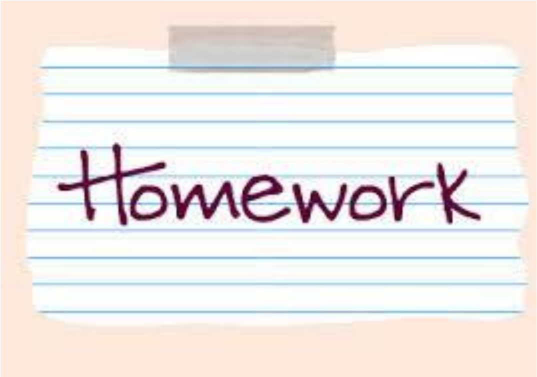
Q 1. What is database?

Q 2. What is SQL?

Q 3. What is Relationship?

Q 4. What is codd’s Rule ?

Q 5. What is record?



Q 1. Explain RDBMS?

Q 2. Write about codd rule?

Q 3. How many types of relationship explain?

Q 4. Explain schema?

Q 5. What is SQL write definition?