Unit-1 Introduction to Database System

Define the following terms.

Data

- Data means known facts that can be stored.
- For example age, salary, result etc...

Database

- A Database is a collection of inter-related Data.
- For example consider a collection of student roll no, name, address, contact number for some student's information. Here all the data - roll no, name, address, contact number are inter- related. They all belong to some particular student.
- Any random collection of data is not considered as a database. For example collections of some vehicle number, balance of account, percentage of student do not form a database.

Management

- Manipulation, searching and security of data
- For example searching of product in amazon, viewing result in GTU website etc...

System

- Programs or tools used to manage database
- For example SQL Server Studio, Oracle 11g

DBMS

- A database management system is a collection of inter-related data and a set of programs to manipulate those data.
- Data manipulation involves various operations such as store data, modify data, remove data and retrieve data.
- DBMS = Database + A set of programs

Metadata

- Metadata is data about data. Data such as table name, column name, data type, authorized user, user access privileges for any table is called metadata for that table.

Data dictionary

- Data dictionary is an information repository which contains metadata. It is usually a part of the system catalog.
- Data dictionary contains description of schema, i.e. overall logical structure of the database. This can involves information such as table names, owners, column names, data types, size and constraints.
- A data dictionary contains the following components:

Entities: An entity is a thing or object or person in the real world that is distinguishable from all other object. E.g. book, student, employee, college etc...

Attribute: Attribute is a property or characteristic of an entity. E.g. entity is student and attributes of students are enrollment no, name, address, cpi etc ...

Relationship: Relationship is an association (connection) between several entities. E.g. book is issued by student where book and student are entities and issue is relation.

Key: A data item or a field which is used to identify a record in a database is referred as key. A primary key is used to uniquely identify a record. For example enrollment no.

Data warehouse

- Data warehouse is an information repository which stored data. It is design to facilitate reporting and analysis.

Data Items (Field)

- A field is a character or group of characters (alphabetic or numeric) that have a specific meaning.
- It is also called a data item. It is represented in the database by a value.
- For Example customer id, name, society, and city are all fields for customer Data.

Record

- A record is a collection of logically related fields.
- Here, each field in a record contains a fixed size and fixed data type.
- For examples, collection of fields — id, name, society & city — forms a record for Customer.
- A record contains values for each field.

Files

- A file is collection of related records.
- These records are generally arranged in a specific sequence.

Differentiate the Data and Information.

Data	Information
Data means known facts, that can be recorded and have implicit meaning	Information means processed or organized data.
Examples: • Student no: 7001 • Student name: Ram • City name: Rajkot • Account No: A01 • Balance: 5000	Examples: • Percentage : 82.20% (Derived from marks of all subject) • Run rate in cricket match: 6.0 run/over (Derived from total runs and over)
Data are row materials used to derive Information.	Information is a product derived from Data.
Data is comparatively less useful.	Information is comparatively more useful.

Differentiate the Schemas and Instances.

| Schema	Instances
The overall logical design of the database is called schema.	The collection of information stored in the database at a particular moment is called instances.
Schema includes table name, column name, data types and size of columns, various constraints at logical level.	Instances include actual data or information stored in table in form of different records or row.
Schema changes infrequently.	Instance change frequently.
Create/Drop of table or columns; changes in data types, size or constraint on any column.	Insert, delete or update operation on data stored in database.

- Sub-schema (External schema) represents part of a database to the end user.

Explain three levels ANSI SPARC Database Architecture.   OR

Explain three level Data abstraction.

The ANSI SPARC architecture divided into three levels:

1) External level (View level)
2) Conceptual level (Logical level)
3) Internal level (Physical level)
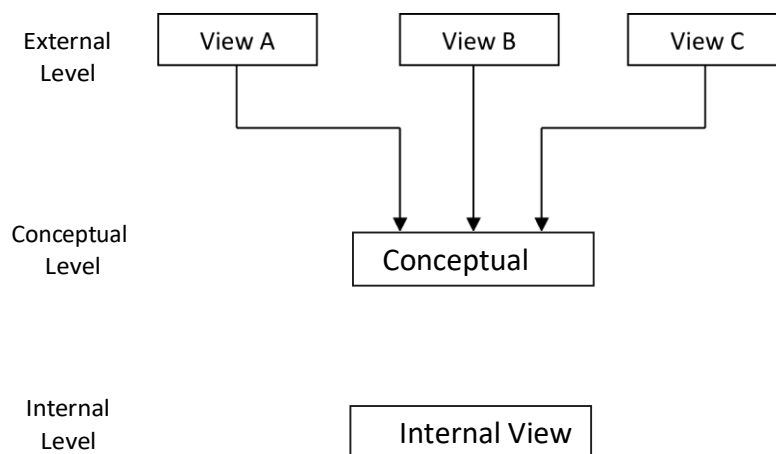
Internal Level

- This is the lowest level of the data abstraction.
- It describes how the data are actually stored on storage devices.
- It is also known as a physical level.
- It provides the internal view of the physical storage of data.
- It deals with data structure, file structure and access methods.
- It also deals with data compression and encryption techniques.

Conceptual Level
- This is the next higher level of the data abstraction.
- It describes what data are stored in the database and what relationships exist among those data.
- It is also known as a logical level.
- DBA and designers work at this level. Application developers also work at this level. They use some pre-defined data types and data-structure to develop applications.

External Level
- This is the highest level of data abstraction. It is also known as view level.
- It describes only part of the entire database that a particular end user requires.
- It provides end users simple interaction with the system.
- It hides details about data types and data-structure used to develop application at logical level.
- End users need to access only part of a database rather than entire database.
- For example, customers need to access only their own account related information.
- Different users need different views of database.

Advantages of Three-tier Architecture
- Same data can be accessed by different users with different customizes views.
- The user is not conserved about the physical data storage details.
- Physical storage structure can be changed without requiring changes in user's view.
- Conceptual structure of the database can be changed without affecting end users.

Explain DDL, DML, DCL and DQL.   OR
Describe component of SQL.
- SQL stands for Structured Query Language.
- SQL is a standard language for accessing and manipulating databases.

DDL (Data Definition Language)
- It is a set of SQL commands used to create, modify and delete database objects such as tables, views, indices, etc.
- It is normally used by DBA and database designers.
- It provides commands like:

   CREATE: to create objects in a database.

   ALTER: to alter the schema, or logical structure, of the database.

   DROP: to delete objects from the database.

   TRUNCATE: to remove all records from the table.

DML (Data manipulation Language)

- It is a set of SQL commands used to insert, modify and delete data in a database.
- It is normally used by general users who are accessing database via pre-developed applications.
- It provides commands like:

  INSERT: to insert data into a table.

  UPDATE: to modify existing data in a table.

  DELETE: to delete records from a table.

  LOCK: to lock tables to provide concurrency control among multiple users.

DQL (Data Query Language)
- It is a component of SQL that allows data retrieval from the database.
- It provides command like **SELECT.** This command is a heart of SQL, and allows data retrieval in different ways.

DCL (Data Control Language)
- It is set of SQL commands used to control access to data and database. Occasionally DCL commands are grouped with DML commands.
- It provides commands like:

  COMMIT: to save work permanently.

  ROLLBACK: to undo work and restore database to previous state.

  SAVEPOINT: to identify a point in a transaction to which work can be undone. GRANT: to give access privileges to users on the database.

  REVOKE: to withdraw access privileges given to users on the database.

Describe different data types in SQL.

- Oracle supports a set of basic data types.
- There are four basic data type available in SQL
    - Numerical
    - Binary
    - Character
    - Date

Numerical Data Types:

Used to store zero, negative and positive numerical values. These values can be fixed-point (whole numbers) or floating-point (real numbers).

No	Data Type	Represent
1	NUMBER(P,S)	Floating-point number. P: precision, i.e. maximum number of digits in a number. Precision can be up to 38 digits. S: scale, i.e. number of digits to the right of the decimal point. Ex: number(6,2) = 1234.79
2	NUMBER(P)	Fixed-lenght number. Ex: number(6) = 123456
3	NUMBER	Floating-point number with a precision of 38 digits.

Character/String data types

No.	Data Type	Description
1	CHAR(size)	Stores character string of fixed length. Size represents the number of characters to be stored ✓ Default size is 1. Maximum length is 255 characters.
2	VARCHAR(size)/ VARCHAR2(size)	✓ Stores character string of variable length. ✓ More flexible than CHAR. No default size will be considered. So, size must be specified ✓ explicitly. ✓ Maximum length is 2000 characters.
3	LONG	✓ Stores large amount of character strings of variable length. ✓ Maximum length is up to 2 GB. Only one column per table can be defined as LONG.

Binary Data Types:

Examples of data which comes under binary type are images, audio, and video files.

No.	Data Type	Description
1	RAW	Stores binary type data. Maximum length is up to 255 bytes.
2	LONG RAW	Stores large amount of binary type data. Often referred as binary large object. Maximum length is up to 2 GB.

Date Data types
- ✓ Used to store date and time.
- ✓ The standard format is DD-MON-YY to store date, such as 1-JAN-11. The current date and
- ✓ time can be retrieved using function SYSDATE.
- ✓ Addition and subtraction operation are possible using number constants and other dates. For example, SYSDATE + 7 will add 7 to current date.

CREATE TABLE QUERY:
This statement is used to create a new table.
Syntax:
CREATE TABLE TABLENAME (COLUMNNAME1 DATATYPE (SIZE),   .., (SIZE));

Example:

CREATE TABLE STAFF_DETAILS (STAFF_ID NUMBER, STAFF_NAME VARCHAR2(20), STAFF_INITIAL VARCHAR2(5), STAFF_MNO INT(10), STAFF_ADDRESS VARCHAR2(30), STAFF_SALARY INT(6), STAFF_HIREDATE DATE,STAFF_TYPE VARCHAR2(10)) ;

DESCRIBE TABLE QUERY:
This statement is too used verify whether table has been created according to specification.
Syntax:
DESC tableName;
Example:
DESC STAFF_DETAILS;

Transaction control and DCL

- ➢ Transaction control (**Transaction Control language)**
  - • A transaction is a unit of work that is performed against a database.
  - • A transaction is the propagation of one or more changes to the database.
  - • For example, if you are creating a record or updating a record or deleting a record from the table then you are performing transaction on the table.
  - • It is important to control transactions to ensure data integrity and to handle database errors.

- Practically you will club many SQL queries into a group and you will execute all of them together as a part of a transaction.
- **Transaction Control:**

There are following commands used to control transactions:

COMMIT: to save the changes.

ROLLBACK: to rollback the changes.

SAVEPOINT: creates points within groups of transactions in which to ROLLBACK

COMMIT Command:
- The COMMIT command is the transactional command used to save changes invoked by a transaction to the database.
- The COMMIT command saves all transactions to the database since the last COMMIT or ROLLBACK command.
- The syntax for COMMIT command is as follows:
- **COMMIT;**

ROLLBACK Command:
- **A ROLLBACK** does exactly the opposite of **COMMIT.**
- The ROLLBACK command is the transactional command used to undo transactions that have not already been saved to the database.
- The ROLLBACK command can only be used to undo transactions since the last COMMIT or ROLLBACK command was issued.
- The syntax for ROLLBACK command is as follows:
- ROLLBACK;

SAVEPOINT Command:
- A SAVEPOINT is a point in a transaction when you can roll the transaction back to a certain point without rolling back the entire transaction.
- The syntax for SAVEPOINT command is as follows:
- SAVEPOINT SAVEPOINT_NAME;
- This command serves only in the creation of a SAVEPOINT among transactional statements.
- The ROLLBACK command is used to undo a group of transactions.
- The syntax for rolling back to a SAVEPOINT is as follows:
- ROLLBACK TO SAVEPOINT_NAME;

DCL (Data Control Language)
Two types of DCL
- Grant
- Revoke

Grant

- Grants a privilege to a user

- It means that giving authority to other user by administrator

- If you are administrator then only you have authority for grating the other authority to other user

- Can grant privilege only if you have been granted that privilege

Syntax:

GRANT < Object Privileges > ON <ObjectName> TO <UserName> [WITH GRANT OPTION];

OBJECT PRIVILEGES

Each object privilege that is granted authorizes the grantee to perform some operation on the object. A user can grant all the privileges or grant only specific object privileges.

The list of object privileges is as follows:

- **ALTER :** Allows the grantee to change the table definition with the ALTER TABLE command

- **DELETE :** Allows the grantee to remove the records from the table with the DELETE command

- **INDEX :** Allows the grantee to create an index on the table with the CREATE INDEX command

- **INSERT :** Allows the grantee to add records to the table with the INSERT command

- **SELECT :** Allows the grantee to query the table with the SELECT command

- **UPDATE :** Allows the grantee to modify the records in the tables with the UPDATE command

Example 1: Give the user rahul permission to only view and modify records in the table client_master**.**

GRANT SELECT, UPDATE ON client_master TO rahul;

Revoke

- **The REVOKE** statement is used to deny the grant given on an object.

- Revokes a privilege from a user

- It is use to taking off or remove of authority or say getting back authority from user

Syntax:

REVOKE < Object Privileges > ON <Object Name> FROM <Username>;

Example 1:

All privileges on the table salesman_master have been granted to rahul. Take back the **Delete** privilege on the table.

REVOKE DELETE ON salesman_master **FROM** Rahul;