
Lab Session 01
(Introduction & Basics)

Database:

A **database** is a collection of information that is organized so that it can be easily accessed, managed and updated.

DBMS:

A **database management system (DBMS)** is system software for creating and managing databases. The **DBMS** provides users and programmers with a systematic way to create, retrieve, update and manage data.

For Example: MySQL, Oracle Database, PostgreSQL, Microsoft Access, Microsoft SQL Server

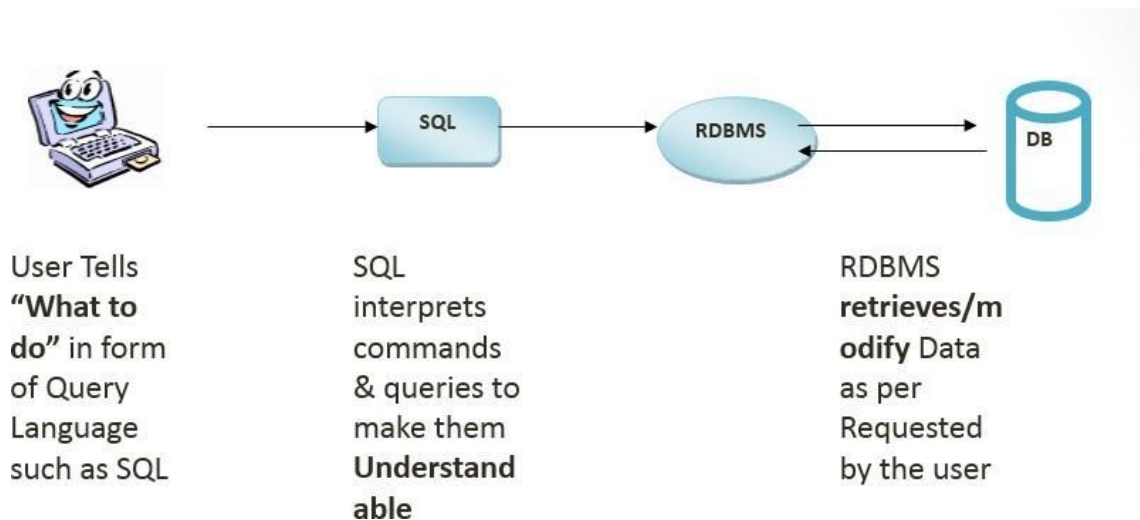
Relational Database:

A **relational database (RDB)** is a collective set of multiple data sets organized by tables, records and columns.

- **Table:** A **table** is a collection of related data composed of rows and columns.
- **Columns:** Columns represent attributes of table.
- **Rows:** It represents single record with in a whole table.

SQL:

SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. It is a non- Procedural Language i.e. User only tell “**What To Do**” not “**How To Do**”. You specify what information you require, rather than how to get it. In other words, SQL does not require you to specify the access methods to the data and provides abstraction.



SQL Categories/Classification:

- **DQL: Data Query Language**
 - Select
- **DML: Data Manipulation Language**
 - Insert, Update, Delete...
- **DDL: Data Definition Language**
 - Create, Alter, Drop, Truncate, Rename
- **DCL: Data Control Language**
 - Grant, Revoke
- **Transaction Control**
 - Commit, Roll Back, Check Point

PL/SQL:

An Oracle Procedural Language which extends SQL by adding application logic.

*

Software Introduction:

Oracle 7 is a relational database management system whereas Oracle 8 and Oracle 9i are object relational database management systems. There are two products Oracle 9i Application Server and Oracle 9i Database that provide a complete and simple infrastructure for internet applications. The Oracle 9i application server (Oracle 9iAS) runs all your applications. The Oracle 9i database stores all your data. In our lab sessions, we will work on latest version of Oracle, Oracle 11g as server and SQL developer as Client.

RETRIEVING DATA USING THE SQL SELECT STATEMENT

The SELECT statement is a DML (Data Manipulation Language) statement. DML statements are SQL commands that allow you to retrieve and manipulate data in the database.

Basic Select Statement

Syntax:

```
SELECT Column_name(s)  
FROM Table_Name
```

Name the
column(s)
to be
displayed

Table
containing
those
columns

The result is stored in a result table; this table is named as **Result Set**

Selecting All Columns

```
SELECT * FROM TABLE_NAME
```

Here '*' (asterisk) means 'All'

Selecting Specific Columns

```
SELECT Column_name1, column_name2  
FROM Table_name where condition ;
```

Name of
the
Column(s)
desired

Conditions In SQL:

- **Between:** Between two values
- **AND:** Both values should be true
- **In (SET):** Among a set of values
- **Like:** Like a character pattern
- **Is Null:** is a null value

Note: Null is a value that is unavailable, unassigned, inappropriate, inapplicable, or unknown. A null is not the same as zero or blank.

Comparison Operators :

Comparison operators are used in conditions that compare one expression to another. They are used in the WHERE or HAVING clause of the SELECT statement.

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

Besides basic comparison operators (>, =, <=, =, <>), Oracle SQL also supports following comparison operators:-

Operator	Meaning
BETWEEN ... AND ...	Between two values (inclusive)
IN (list)	Match any of a list of values
LIKE	Match a character pattern
IS NULL	Is a null value

Logical Operators :

A logical operator combines the result of two component conditions to produce a single result based on them or to invert the result of a single condition. Three logical operators are available in SQL as shown below:-

Operator	Meaning
AND	Returns TRUE if both component conditions are TRUE
OR	Returns TRUE if either component condition is TRUE
NOT	Returns TRUE if the following condition is FALSE

Viewing your Objects

Write the following command to SQL prompt:

SQL > select * from cat;

This command is used to retrieve category objects from a user. You will find no-row in this case. As you have not created any object in your account so far.

Aggregate Functions:

count: used for counting rows

max: maximum value for a

column **min:** minimum value

for a column

sum: computes the sum of values (only applicable to the data type number)

avg: computes average value for a column.

Note: avg, min and max ignore tuples that have a null value for the specified attribute, but count considers null values.

Some Basic Queries!!

1. Retrieve all rows & columns from the emp table.

```
SELECT * FROM emp;
```

// '*' denote all attributes. See other tables.

2.The columns to be selected from a table are specified after the keyword SELECT.

```
SELECT empno, ename, sal FROM emp;
```

3.The SELECT clause may also contain arithmetic expression,

e.g.

```
SELECT ename, deptno, sal * 2 FROM emp;
```

4.Keyword DISTINCT forces elimination of duplicates from the query result.

```
SELECT DISTINCT job FROM emp;
```

5. To sort the result ORDER BY clause is used.

It takes one or more attributes listed in the SELECT clause. Asc specifies ascending order (default order) and desc specifies descending order.

SELECT ename, deptno, hiredate FROM emp ORDER BY deptno asc;

6. List name, job title and the salary of those employees whose manager has the number 7698 or 7566 and who earn more than 1500.

SELECT ename, job, sal FROM emp WHERE (mgr = 7698 or mgr = 7566) and sal > 1500;

Note: To select tuples that satisfy certain condition, WHERE clause is used. WHERE clause condition contains logical expressions connected by and, or and not to form complex conditions.

//Conditions may also include pattern matching and subqueries.

7. List name, job and hiredate of employees working in department number 20 or 30.

SELECT ename, job, hiredate FROM emp WHERE deptno IN (20,30);

8 • List the emp_no and name of employee who has no manager.

• SELECT empno, ename FROM emp WHERE mgr IS NULL;

9 • List emp_no, name and job of employees whose salary is between 1500 – 2500 •

SELECT empno, ename, job FROM emp WHERE sal BETWEEN 1500 AND 2500;

10 • Find name and job of employees whose name contain substring 'LL'.

• SELECT ename FROM emp WHERE ename like '%LL%';

Note: For pattern matching like operator is used. Two special characters are used: the percent sign (%) and under score (_). Percent sign means that any string is allowed there, while under score stands for exactly one character.

11 • Count the number of employees.

• SELECT count(*) as Total_Employees FROM emp;

Note: Aggregate functions return a single value for a set of column values.

EXERCISES:

1. Display the name, job, and salary for all employees whose job is Clerk or Analyst and their salary are not equal to Rs.1000, Rs.3000, or Rs.5000.
2. Display in descending order of salary. (4 rows)
3. Calculate the average salary of all the employees. (2073.21429)
4. List all the employees names, their department numbers and hire date with ascending order of dept numbers and descending order of hire date within a particular dept. (multiple columns in ORDER BY clause).
5. Find emp_no, name, salary and hire date of the employees who were hired in the first half year of 1981.
6. Sort the output in ascending order of hiredate.
7. How many different job titles are stored in the relation emp? (5)
8. List the minimum and maximum salary in relation emp. (800, 5000)
9. Sum all the salaries of employees working in the SALES department. (9400)
10. Compute the difference between the minimum and maximum salary. (4200)
11. List all employees whose name contains A in the third position. (3 names)
12. How many employees earn more than Rs. 2000. (6 emps)

13. Find total number of salesman and the sum of their salaries. (14, 29025)