DATABASE MANAGEMENT SYSTEM
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. A DBMS makes it possible for end users to create, data reading, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

3 important things that DBMS manages are:

a) the data,

b) The database engine that allows data to be accessed, locked and modified, and

c) The database schema, which defines the database's logical structure.

Depending upon the usage requirements, there are following types of databases. They are:

(i) CENTRALISED DATABASE:

The data is stored at a centralised location & the users from different location can access this data.

This type of database contains application procedures that help the user to access the data even from a remote location.

(ii) RELATIONAL DATABASE :

these databases are categorized by a set of tables where data get fit into a pre-defined category. The table consists of rows and columns where the column has an entry for data for a specific category and rows contains instance for the data defined according to the category.

(iii) CLOUD DATABASES :

A cloud database is a database that has been optimized or built for such a virtualized environment. Benefits of cloud database are the ability to pay for storage capacity and bandwidth on a per-user basis, and they provide scalability on demand, along with high availability.

EXAMPLES OF DATABASE MANAGEMENT SYSTEM :

a) ORACLE :

Oracle database is a multi-model database management system produced and marketed by Oracle Corporation.

It is a database usually used for running Online

transaction processing (OLTP), data warehousing, and mixed data. It offers a wide range of options and features in the areas of availability, scalability, Analytics, Performance, security, management, developers and integration.

these aim to enhance and complement existing database functionality to meet customer specific requirements.

Oracle software was developed by the Software Development laboratories (SDL).

b) MySQL :

MySQL database is an open source database mgmt. system which is Relational.

It is written in C and C++. It is a component of the LAMP web application software stack.

MySQL is based on a client - server model. The core of MySQL is MySQL server, which handles all the database instructions (or commands).

the RDBMS supports large databases with million records and supports many data types.

It support development in virtualized environments.

QUERY 1 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP923	EMPNO	Number	-	-	0	✓	✓	-	-
	ENAME	Varchar2	50	-	-		✓	-	-
	JOB	Varchar2	20	-	-		✓	-	-
DEPTNO	Number	-	-	0			-	-

1-4

* QUERY 1 :

Write a query to create following tables :

i) EMP (Empno, Ename, job, deptno)
ii) DEPT (Deptno, Dname, location)
iii) CUSTOMER (Cno, Cname, address, phone)
iv) STUDENT (studentname, studentno, address)

* i) create table EMPO23
(Empno int (20), Ename varchar (50),
Job (varchar (20), Deptno int (10));

ii) create table DEPTO23
(Deptno int (20), Dname varchar (varchar (30),
Location varchar (30));

iii) create table CUSTOMERO23
(Cno int (40), Cname varchar (30),
Address varchar (50), Phone varchar (10));

iv) create-table STUDENTO23
(Studentname varchar (30), Studentno varchar (20),
Address (varchar (50));

QUERY 2 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP023	EMPNO	Number	-	-	0	-	-	-	-
	ENAME	Varchar2	50	-	-	-	✓	-	-
	JOB	Varchar2	20	-	-	-	✓	-	-
	DEPTNO	Number	-	-	0	-	✓	-	-
	PHONENO	Varchar2	10	-	-	-	✓	-	-

1-5

QUERY 3 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP023	EMPNO	Number	-	-	0	-	-	-	-
	ENAME	Varchar2	40	-	-	-	✓	-	-
	JOB	Varchar2	20	-	-	-	✓	-	-
	DEPTNO	Number	-	-	0	-	✓	-	-
	PHONENO	Varchar2	10	-	-	-	✓	-	-

1-5

QUERY 4 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP023	EMPNO	Number	-	-	0	1	-	-	-
	ENAME	Varchar2	40	-	-	-	✓	-	-
	JOB	Varchar2	20	-	-	-	✓	-	-
	DEPTNO	Number	-	-	0	-	✓	-	-
	PHONENO	Varchar2	10	-	-	-	✓	-	-

1-5

* QUERY 2 :
  Write a query to add a new column PHONEND to the
  employee table :

* alter table EMP023
  add Phoneno varchar (10) ;

* QUERY 3 :
  Write a query to modify size of existing attribute
  ENAME :

* alter table EMP023
  modify Ename varchar (40) ;

* QUERY 4 :
  Write a query to add PRIMARY KEY in the tables :

* i) alter table EMP023
     add constraint PK001 primary Key (Empno) ;

  ii) alter table DEPT023
      add constraint PK002 primary key (Deptno) ;

  iii) alter table CUSTOMER023
       add constraint PK003 primary key (Cno) ;

QUERY 5 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMER023	CNO	Number			0				
	CNAME	Varchar2	30				✓		
	ADDRESS	Varchar2	50				✓		
	PHONE	Varchar2	10				✓		

1 - 4

QUERY 6 :

> Table dropped.
> 0.23 seconds

QUERY 7 :

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP023	EMPNO	Number			0	1			
	ENAME	Varchar2	50				✓		
	JOB	Varchar2	20				✓		
	DEPTNO	Number			0		✓		
	PHONENO	Varchar2	10				✓		

1 - 5

QUERY 8 :

> Table altered.
> 0.00 seconds

iv) alter table STUDENT023
add constraint PK004 primary key (studentno);

* QUERY 5 :
Write a query to describe table customer :

* desc CUSTOMER023 ;

* QUERY 6 :
Write a query to drop table student :

* drop table STUDENT023 ;

* QUERY 7 :
Write a query to add a constraint in emp table that the
name should not be null :

* alter table EMP023
modify (Ename varchar(50) not null);

* QUERY 8 :
Write a query to add a constraint in the table EMP023 that
the value of Empno should be greater than 0 :

* alter table EMP023
add check (Empno > 0);

ENAME	JOBTYPE	ENO
Shreyansh	Manager	1
Abhay	Manager	2
Vayu	Clerk	3
Vishal	Peon	4

JOBTYPE
Manager
Clerk
Peon

ENAME	SALARY
Shubham	60000

* QUERY 1 :
  Write a query to display employname, jobtype, empno
  for each employee in the employee table :

* select ENAME, JOBTYPE, ENO
  from EMPLOYEE023;

* QUERY 2 :
  Write a query to display unique job for the employee
  table :

  select unique (JOBTYPE) from EMPLOYEE023;

* QUERY 3 :
  Write a query to display employeename and salary of the
  employees earning > 50000 :

  select ENAME, SALARY from EMPLOYEE023
  where SALARY > 50000;