DATA BASE MANAGEMENT SYSTEM

SQL

In Relational database Model all the information is stored on Tables, these tables are divided into rows and columns. A collection on related tables are called DATABASE. A named table in a database is called RELATION in Relational Data Model. Row in a table is called TUPLES, and column of a Table are called ATTRIBUTE. No. of rows in a table is called DEGREE of the table and no. of columns in a tables in called CARDINALITY.

Primary Key : An attribute or a group of attribute which can distinguish a row uniquely in a table is Called Primary key/Key Field/Key attribute.
Candidate key : The attributes which are capable to act as a primary key is known as candidate key.
Alternate key : An attribute which can act as a primary key in place of primary key as called alternate key.
Foreign Key : An attribute in it's present table whose values are derived from some other table, is called foreign key in the present table.

SQL stand for structured query language, a language to manipulate database. Now a days It is a universal database language to create , manipulate database across plate forms.

The data types supported by the standard SQL are as follows

S.No	Data Type	Meaning	Example
1	Char	It can store information containing alphabets and numeric or alphanumeric	**Char (30)** can store "ram" , "ramji007" or "80- b surya Nagar
2	Vhar2	It can store information containing alphabets and numeric or alphanumeric. But these data types when used can increase it 's size when required	**Varchar2 (30)** can store "ram" , "ramji007" or "80- b surya Nagar
3	Number	It can store numeric values which can have fractional part or without fractional part	(1) **Number(2)** mean it can store a numeric values between 00 and 99 (2) **number(8,2)** means it can store values whose after decimal points values can go upto two places
4	Date	It is used to store date type information	**Date** can store any valid date

Operator : SQL supports different types of operators some of them is as follows

1. **Arithmetic operator** Example are : + , - , * and /
2. **Logical operator** examples are : > , < , =, >= , <=, != (not equal to)
3. **Relational Operator** Examples area : AND , OR , NOT

DDL : DDL stand for data definition language . it is basically responsible to define the structure of any database table. The commands falls under this category is as follows

1. Create Table
2. Alter Table
3. Drop Table

DML : DML stand for data manipulation language. This is basically responsible for managing the records(row/tuple) of any table. The main command falls under this category are

1. Insert
2. Delete
3. Update
4. Select

5. Create view
6. Drop View

Create Table : This DDL command is used to create a new table into any existing database . The syntax of this command is as follows

Syntax

Create table <tblname> (column_name datatype size constraints , column_name datatype size constraints , column_name datatype size constraints , ----------
----
);

Example :

Sql>Create table student (admno number(4) , roll number(2),
Name char(30), fname char(30),
DOB date
Address char(80)
);

To see the structure of the above defined table is as follows

Sql> desc student ;

ALTER TABLE : This DDL command is used to add / modify a column in any existing table of a database The syntax of this command is as follows

Syntax

Alter table <tblname> ADD/MODIFY/DROP (column_name datatype size constraints ,column_name datatype size constraints ,column_name datatype size constraints , ----------
----------);

Example

Task : To add a new column " Phone " having datatype char and size 12 in a table student

Sql > alter table student ADD (phone char (12)) ;

Task : To change the datatype of admno into char and increase the size as 5

Sql > alter table student MODIFY (admno char(5));

Task : Delete a column DOB from the table student

Sql > alter table student drop DOB;

Drop Table ; This DDL command is used to remove a table from any existing database. This syntax of this command is as follows

Syntax: Drop table < table name>

Example : Sql> drop table student ;

INSERT COMMAND : This DML command is used to add a new row(record/ tuple) in any existing database table . The syntax is as follows

Syntax: insert into < table name > values (value1, value2, value 3,.....value N);

Example Task: Add a new row in a table student (which we have created earlier)

Sql > Insert into student values ('1234A','12,' 'joshin gulati',''abcd', '9-apr-2008','b-152 surya nagar','0120-2626132'

Note : Always put char type values and date type values in single inverted comma

SELECT COMMAND : This DML command is used to retrieve information from any table (s) to display/ list/ report on the screen, but this command can not any how update/ modify any table. The syntax of this command is as follows

Syntax: Select [* / column list] from < table name>
[where < condition >]
[order by < col name..> [asc/desc]]
[group by < col name>]

Examples

Task : To display all the records of table EMP *Sql > select * from EMP;*

Task : To display only the employee no and employee name *Sql > select empno , ename from EMP;*

Task : to display employee name in first column and employee number on second column
Sql> select ename, empno from EMP;

Where clause : it is used to restrict the no no of rows from being displayed on the screen

Task : display all the employees whose salary is greater than 1500 Sql > select * from emp where sal > 1500

Task : display all the employees information whose salary is between 1500 and 2500
Sql > select * from emp where sal > =500 and sal <=2500

Task : Display all the information of clerks Sql > select * from emp where job ='CLERK';

Task :display the information of all the employees whose job is 'salesman ' or 'CLEAK'
Sql> select * from emp where job ='SALESMAN ' OR job ='CLERK'

Task : Display the information of all those employee whose job is not president
Sql> select * from emp where job !='PRESIDENT' OR
Sql> select * from emp where NOT job='PRESIDENT'

Calculated field :

Task : Display employee number, name and salary earned by each employee in a year from the table EMP (Suppose salary is paid in every month) Sql > select empno,ename , salary *12 from EMP;

NOTE : In the above example salary * 12 is a calculated field.

Comparing with NULL values Task : Display employees information who earns NULL commission
Sql > select * from EMP Where sal is NULL

NOTE : Do not use equal sign to compare NULL values.

Like Clause -----Pattern Matching : This clause is used to display only those records which contains a single or multiple char from a given set of values .

Task : Display all the names starting with alphabet 'A' from table EMP.
Sql > select ename from emp where ename like 'A%;

Task : display all the name starting with alphabet 'A' and must contains only 5 letters
Sql> select ename from emp where ename like 'A_ _ _ _';

Note : % --- > replace all the remaining char(s)
_ (under Score) ----- > Replace only a single char at a time

Removing Duplicate Rows (Distinct Keyword)

Task : Display Different jobs available in Table EMP Sql > select distinct (job) from EMP;

Order By clause : This select clause is used to display the retrieved data in an arrange format

Task : display employee table information according to the name in asceding order from the table EMP
Sql > select * from emp order by name asc

Task : Display only employee names in descending order from the table EMP
Sql > display ename from emp order by ename desc

Task : Dispay the information of only clerk in ascending order according to their name from table EMP
Sql > select ename from EMP Where job ='clerk' Order by ename asc;

NOTE : Asc is not compulsory but desc is compulsory

Grouping Function: These functions are operative on the whole table, so the result of these functions are based on the whole table. These functions can not be applied on a single record/row

1. **AVG( ) : try to find out average of given data--------Applicable on numeric only**
2. **Sum( ) : Try to find out sum of given data ----------Applicable on numeric only**
3. **Count :** count total no of rows of a table ……**-Applicable on all type of data**
4. **Min : find out the minimum numeric values …….-Applicable on numeric only**
5. **Max : : find out the maxi numeric values…… …….-Applicable on numeric only**

Example :

Task : find out Maximum salary paid Sql> select max(sal) from EMp;
Task : find out minimum salary paid Sql > select min(sal) from emp;
Task : find out average salary paid to all employees Sql > select avg(sal) from EMP;
Task : find out total salary paid to all employees Sql > select sum (sal) from EMp;
Task : find out total no of employee Sql > select count(*) from emp;

Task : find out total no of different types of job from the table EMp

Sql > Select count(distinct job) from emp;

Group By Clause Example

Task : Find out the total no of employee in each jobs from EMP table

Sql > select job , count(job) from EMP Group by job;

Task : Find out average salary of clerks from table EMP

Sql > select job , avg(sal) from EMP Where job ='CLERK' Group By job

Joins

Joins : A join is a query that combines rows from two or more than two tables. The function of combining Data from multiple tables is called joining. In a join query more than one table is listed in FROM Clause.

Example Sql > Select * from EMP, DEPT;

NOTE : when two tables are joined without any condition then the Numbers of columns in the resultant query is the addition of first table and second tables, and the number of row are the multiplication of both tables row --- This is also called **Cartesian product** of two tables)

EQUI JOIN Sql > select * from emp , dept

Where emp.deptno = dept. deptno

(The above two tables are joined by some condition ---This is called equi Join – but one column appear twice)

Natural JOIN Sql > select empNp, ename , emp.deptno from emp , dept

Where emp.deptno = dept. deptno **(In this case duplicate column appears only once)**

Update : This DML command is used to change/modify the record(s) of any table. The syntax is as follows

Syntax:

```
Update  < tblname>
        Set    column Name = value
[ where < condition> ]
```

Example

1. Update emp
   Set sal = sal + sal * 0.05; (Give an increment of 5 % to each employee)
2. update emp
   set sal = sal+ sal * 0.05
   where job ='PRESIDENT'; (Give an increment of 5% to president only)

Delete : This DML command is used to delete a row/ tuple(s) from a table. The Syntax is as follows

Syntax

```
DELETE from <tblname >
        [ where < condition > ]
```

Example

1. Delete from emp ; (delete all the records leaving it's structure intact)
2. delete from emp where sal >=3000 ;

Create View : This Command is used to create a new View in any existing database. The syntax is as Follows

Syntax

```
Create view < viewname>
        As
Select command
```

Example

```
create view abc  as select  empno,job,sal from emp;
```

Drop View : This command is used to drop any existing view from the database. The syntax is as follows

Syntax

Drop View < viewname>

Example

Drop view abc;

ALTER TABLE COMMAND:- It is used to change the definitions of existing tables. Usually , it can add columns or change their sizes.

ADD COLUMN :- syntax:-

Alter table <table name>
Add <columnname> <datatype><size>;

Exam:- Alter table student1
Add city varchar2(20);

MODIFY COLUMN :- syntax:-

Alter table <table name>
Modify <columnname> <datatype><size>;

Exam:- Alter table student1
Modify city varchar2(25);

Views:- A view is a (virtual) table that does not really exist in its own right but is instead derived from one or more base table(s).

Syntax:- Create view <viewname>
AS (select statement);

Exam:- Create view view1
AS (select name, age, fee from student1);

DROP VIEW COMMAND:- The Drop view command is used to delete a view from the database.

Syntax:- Drop view <viewname>;

Exam:- Drop view view1;

CONSTRAINT :- A constraint is a condition or check applicable on a field or set of fields.

Syntax:- Create table < table name>
(<column name> <data type> <size> <column constraint>,
<column name> <data type> <size> <column constraint>,
<column name> <data type> <size> <column constraint> …..);

1. Default constraint:- A default value can be specified for a column using default clause . when a user does not enter a value for the column , automatically the defined default value is inserted in the field.

2. Check Constraint:- This constraints limits values that can be inserted into a column of a table.

3. Unique Constraint:- This constraint ensures that no two rows have the same value in the specified column(s).

4. Primary key :- This constraint declares a column as the primary key of the table. This constraint is similar to unique constraint except that only one column can be applied in this constraint.

The primary key cannot allow NULL values.

5. NOT NULL:- This constraint ensures that the column cannot be left empty.

Exam:- Create table emp
(empno number(3) primary key,
empnaem vharchar2(20) unique key
city varchar2(20) default =' delhi',
age number(3) check age>30,
salary number (7,2) NOT NULL);

Q1.(a) Write an Sql query to create the table 'Menu' with the following structure:

Field	Type	Constraint
ItemCode	Varchar(5)	Primary Key
ItemName	Varchar(20)	
Category	Varchar(20)	
Price	Decimal(5,2)	

(b) Write an Sql query to create the table 'TEAMS' with the following structure:

Field	Type	Constraint
TeamCode	Varchar(5)	Primary Key
TeamName	Varchar(20)	
TeamLeader	Varchar(20)	
NoOfMembers	Integer	
Team_Symbol	Char(1)	Not Null

(c) Write an Sql query to create the table 'Voter' with the following structure:

Field	Type	Constraint
Voter_No	Varchar(10)	Primary Key
Voter_Name	Varchar(30)	Not Null
Address	Varchar(40)	
Age	Integer(3)	
City	Varchar(15)	
State	Varchar(15)	

(d) Write an Sql query to create the table 'Sports' with the following structure:

Field	Type	Constraint
Sno	Integer(3)	Primary Key
Name	Varchar(30)	Not Null
Age	Integer(2)	
Address	Varchar(30)	
City	Varchar(20)	

(e) Write an Sql query to create the table 'Bank' with the following structure:

Field	Type	size	Constraint
Acc_Number	Integer	4	Primary Key
Name_Number	Varchar	20	
Birth_Date	Date		
Balance_Amount	Integer	8	Not Null

Q1. Write SQL commands for Create table ***Teacher*** **a**nd insert data in table.

No	Name	Age	Department	Date of Join	Salary	Sex
1.	jigal	34	Computer	10/01/97	12000	M
2.	Sharmila	31	History	24/03/98	20000	F
3.	Sandeep	32	Maths	12/12/96	30000	M
5.	Rakesh	42	Maths	05/09/97	25000	M
6.	Shyam	50	History	27/02/97	30000	M
7.	Shiv Om	44	Computer	25/02/97	21000	M
8.	Shalakha	33	Maths	31/07/97	20000	F

1. Sorting Result- ORDER BY Clause:

SELECT <column-name>[,<column-name>…….] FROM <table name> WHERE <condition> ORDER BY <column-name>

2. The INSERT Command: INSERT INTO <table-name>[<column list>] VALUES (<value>,<value>,<value>,…..);
3. The DELETE Command: DELETE FROM <table-name> WHERE <condition>; DELET FROM<table-name>;
4. The UPDATE Command: UPDATE emp SET sal=5000 WHERE empno=1011;
5. The ALTER TABLE Command: (to add a column to a table):

ALTER TABLE <table-name> ADD <column-name> <data type> <size>;

6. Syntax:ALTER TABLE <table-name> MODIFY (Columnname newdatatype (newsize));
7. The DROP CommandDROP TABLE <table-name>
8. Like Clause -----Pattern Matching : This clause is used to display only those records which contains a single or multiple char from a given set of values .

Task : Display all the names starting with alphabet 'A' from table EMP.

Sql> select ename from emp where ename like 'A%;

Task : display all the name starting with alphabet 'A' and must contains only 5 letters

Sql> select ename from emp where ename like 'A_ _ _ _';

Note : % --- > replace all the remaining char(s)

_ (under Score) --- > Replace only a single char at a time

EQUI JOIN Sql>select * from emp , dept Where emp.deptno = dept. deptno;

(The above two tables are joined by some condition ---This is called equi Join – but one column appear twice)

9. Natural JOIN Sql>select empNp, ename , emp.deptno from emp , dept Where emp.deptno = dept. deptno

(In this case duplicate column appears only once)

10. Group By Clause : The rows of a table can be grouped together based on a common value by using the Group By clause of SQL in a select statement.
11. Syntax : SELECT <attribute name>, <attribute name> ---- [functions] FROM <relation name> GROUP BY <group by column>;

Eg. Select age, count (rollno) From students Group by age;

Output : Age Count(rollno)

15 2

14.5 2

12. Group-By-Having Clause : It is used to apply some condition to the Group By clause.

Eg. Select class From students Group by class Having count(*) > 5;

Write SQL commands for all on the basis of Teacher relation given below.

No	Name	Age	Department	Date OF Join	Salary	Sex
1.	Jigal	34	Computer	10/01/97	12000	M
2.	Sharmila	31	History	24/03/98	20000	F
3.	Sandeep	32	Maths	12/12/96	30000	M
4.	Sangeeta	35	History	01/07/99	40000	F
5.	Rakesh	42	Maths	05/09/97	25000	M
6.	Shyam	50	History	27/02/97	30000	M
7.	Shiv					
8.	Om	44	Computer	25/02/97	21000	M
9.	Shalakha	33	Maths	31/07/97	20000	F

1. To show all information about the teacher of history department.
2. To list the names of female teachers who are in Maths department
3. To list names of all teachers with their date of joining in ascending order.

4. To display teachers name, age department and salary for male teacher only
5. To count the number of teachers with age>23.
6. To insert a new row in the teacher table with the following data: 9,"raja",26,"computer",13/05/95,2300,"m".
7. To insert a new row in the teacher table with the following data: 19,"amit kumar".
8. To show all information about the teachers in this table
9. Add a new column named "Address" with size 50
10. Arrange the whole table in the alphabetical order to name
11. Display the age of the teachers whose name starts with 'S'.
12. Display name and age in ascending order using age
13. Count number of departments present in table
14. Count number of record's present in table
15. Display sum of all salaries
16. Display avg of all salaries
17. Display min and max salary
18. Display all record whose sex female
19. Display all name in capital letters.
20. Display age between 35 to 50
21. Display distant departments
22. Display distant age
23. Display distant salary
24. Count number of teachers from math's dept
25. Update age by 2 which age grater then 40.
26. Display all record where name start with 'O'
27. Display all record where name start with 'R'
28. Display name and age in descending order.
29. Give the output of following statement.
30. select count(distinct department) from teacher.
31. select max(age)from teacher where sex="f"
32. select avg(salary) from teacher where dateofjoin<12/07/96
33. select sum(salary) from teacher where dateofjoin<12/07/96
34. select sysdate from dual;
35. select name from teacher where name like "%a";
36. select min(salary) from teacher;
37. select department, count(*) from teacher where count(*)<2;
38. select name, age ,sex from teacher where age >45;
39. select distinct(salary) from teacher group by department;
40. select distinct(age) from teacher group by age;
41. select distinct(sex) from teacher group by sex;

ANSWERS:

1) To show all information about the teacher of history department.
   Select *from teacher where department= "history";
2) To list the names of female teachers who are in Maths department
   Select *from teacher where department= "Maths";
3) To list names of all teachers with their date of joining in ascending order.
   Select name, Date_of_Join from teacher order by Date_Of_Join;
4) To display teachers name, age department and salary for male teacher only
   Select name, age, department, salary from teacher where sex='M';
5) To count the number of teachers with age>23.
   Select count(*) from teacher where age>23;
6) To insert a new row in the teacher table with the following data: 9,"raja",26,"computer",13/05/95,2300,"m".
   Insert into teacher values(9,"raja",26,"computer",13/05/95,2300,"m");
7) To insert a new row in the techer table with the following data: 19,"amit kumar".
   Insert into teacher(Age,Name) values(19,"amit kumar");

8) To show all information about the teachers in this table
   Select *from teacher;
9) Add a new column named "Address" with size 50
   Alter table teacher add Address int(50);
10) Arrange the whole table in the alphabetical order to name
   select Avg(salary) from teacher order by name;
11) Display the age of the teachers whose name starts with 'S'.
   Select age from teacher where name like "S%";
12) Display name and age in ascending order using age
   Select Name,Age from teacher order by age ;
13) Count number of departments present in table
   Select Count(Departments) from teacher;
14) Count number of record's present in table
   Select Count(*) from teacher ;
15) Display sum of all salaries
   Select Sum(Salary) from teacher ;
16) Display avg of all salaries
   Select Avg(Salary) from teacher ;
17) Display min and max salary
   Select MIN(Salary), MAX(Salary) from teacher ;
18) Display all record whose sex female
   Select * from teacher where sex="F" ;
19) Display all name from department of History ,Maths .
   Select Name from teacher where department in("History" ,"Maths");
20) Display age between 35 to 50
   Select Age from teacher where age between (35,35);
21) Display distant departments
   Select distinct(department) from teacher;
22) Display distant age
   Select distinct(age) from teacher;
23) Display distant salary
   Select distinct(salary) from teacher;
24) Count number of teachers from math's dept
   Select coutn(*) from teacher where department="Maths";
25) Update age by 2 which age grater then 40.
   Update emp SET Age=Age+2 Where age >40;
26) Display all record where name start with 'O'
   Select *from teacher where name like "O%";
27) Display all record where name start with 'R'
   Select *from teacher where name like "R%";
28) Display name and age in descending order.
   Select Name ,Age from teacher order by age desc;
29) GIVE THE OUTPUT OF FOLLOWING STATEMENT.
30) select count(distinct department) from teacher.
   3
31) select max(age)from teacher where sex="f"
   40000
32) select avg(salary) from teacher where Date_of_Join<12/07/96

33) select sum(salary) from teacher where dateofjoin<12/07/96
select sysdate from dual;
current date

34) select name from teacher where name like "%a";
Sharmila
Sangeeta
Shalakha

35) select min(salary) from teacher;
12000

36) select department, count(*) from teacher where count(*)<2;
Maths 3
History 3

37) select name, age ,sex teacher where age >45;
Shyam 50 M

38) select distinct(salary) from teacher group by department;
12000
20000
30000
40000
21000

39) select distinct(age) from teacher group by age;
34
31
32
35
42
50
44
33

40) select distinct(sex) from teacher group by sex;
M
F