

cities from table suppliers.

SELECT DISTINCT City FROM Suppliers;

12. Consider a database LOANS with the following table:

Table: LOANS

AccNo	Cust_Name	Loan_Amount	Instalments	Int_Rate	Start_Date	Interest
1	R.K. Gupta	300000	36	12.00	2009-07-19	1200
2	S.P. Sharma	500000	48	10.00	2008-03-22	1800
3	K.P. Jain	300000	36	NULL	2007-03-08	1600
4	M.P. Yadav	800000	60	10.00	2008-12-06	2250
5	S.P. Sinha	200000	36	12.50	2010-01-03	4500
6	P. Sharma	700000	60	12.50	2008-06-05	3500
7	K.S. Dhall	500000	48	NULL	2008-03-05	3800

Answer the following questions.

- Display the sum of all Loan Amounts whose Interest rate is greater than 10.
- Display the Maximum Interest from Loans table.
- Display the count of all loan holders whose name ends with 'Sharma'.
- Display the count of all loan holders whose Interest rate is Null.
- Display the Interest-wise details of Loan Account Holders.
- Display the Interest-wise details of Loan Account Holders with at least 10 instalments remaining.

Ans. (i) Mysql> SELECT SUM(Loan_Amount) FROM LOANS WHERE Int_Rate >10;
(ii) Mysql> SELECT MAX(Interest) FROM LOANS;
(iii) Mysql> SELECT COUNT(*) FROM LOANS WHERE Cust_Name LIKE '%Sharma';
(iv) Mysql> SELECT COUNT(*) FROM LOANS WHERE Int_Rate IS NULL;
(v) Mysql> SELECT * FROM LOANS GROUP BY Interest;
(vi) Mysql> SELECT * FROM LOANS GROUP BY Interest HAVING Instalments>=10;

the above query

For example,

```
SELECT * FROM Student ORDER BY City DESC;
```

It displays all the records of table student ordered by city in descending order.

GROUP BY clause

The GROUP BY clause can be used in a SELECT statement to collect data across multiple records and group the results by one or more columns.

For example,

```
SELECT Name, COUNT(*) AS "Number of employees"
```

```
FROM Employee WHERE Salary>35000.
```

```
GROUP BY City;
```

It displays name and the total number of employees of each city who are getting salary greater than 35000.

15. Consider the following tables Product and Client. Write SQL commands for the statements (i) to (iii) and give outputs for SQL queries (iv) to (vi).

Table: PRODUCT

P_ID	Product Name	Manufacturer	Price
TP01	Talcum Powder	LAK	40
FW05	Face Wash	ABC	45
BS01	Bath Soap	ABC	55
SH06	Shampoo	XYZ	120
FW12	Face Wash	XYZ	95

Table: CLIENT

C_ID	Client Name	City	P_ID
01	Cosmetic Shop	Delhi	FW05
06	Total Health	Mumbai	BS01
12	Live Life	Delhi	SH06
15	Pretty Woman	Delhi	FW12
16	Dreams	Bengaluru	TP01

- (i) To display the details of those Clients whose city is Delhi.
 - (ii) To display the details of Products whose Price is in the range of 50 to 100 (both values included).
 - (iii) To display the details of those products whose name ends with 'Wash'.
 - (iv) `SELECT DISTINCT City FROM CLIENT;`
 - (v) `SELECT Manufacturer, MAX(Price), MIN(Price), COUNT(*) FROM PRODUCT GROUP BY Manufacturer;`
 - (vi) `SELECT Product Name, Price * 4 FROM PRODUCT;`
- ns. (i) `SELECT * FROM CLIENT WHERE City="Delhi";`
- (ii) `SELECT * FROM PRODUCT WHERE Price BETWEEN 50 and 100;`
- (iii) `SELECT * FROM PRODUCT WHERE ProductName LIKE '%Wash';`

16. Define a Foreign Key.
 Ans. A foreign key is a key which is used to link two tables together. It is also called a referencing key. Foreign key is a column or a combination of columns whose values match a primary key in a different table. The relationship between two tables matches the primary key in one of the tables with a foreign key in the second table. If a table has a primary key defined on any field(s), then you cannot have two records having the same value of that field(s).

17. Define the various SQL Constraints.
 Ans. Constraints are the rules enforced on data or columns on a table. These are used to restrict the values that can be inserted in a table. This ensures data accuracy and reliability in the database. Following are the most commonly used constraints available in SQL:

- (a) NOT NULL Constraint: Ensures that a column cannot have NULL value.
- (b) DEFAULT Constraint: Provides a default value for a column when no value is specified.
- (c) UNIQUE Constraint: Ensures that all values in a column are unique. There should not be any redundant value in a column which is being restricted.
- (d) PRIMARY Key: Uniquely identifies each row/record in a database table.
- (e) FOREIGN Key: Uniquely identifies a row/record in any other database table.
- (f) CHECK Constraint: The CHECK constraint ensures that all values in a column satisfy certain conditions. For example, to restrict the salary column that it should contain salary more than ₹ 10,000.

18. Write the outputs of the SQL queries (i) to (iii) & queries for (iv) to (viii) based on the relations Teacher and Posting given below:
 [CBSE Sample Paper 2020-21]

Table: Teacher

T_ID	Name	Age	Department	Date_of_Join	Salary	Gender
1	Jugal	34	Computer Sc	2017-01-10	12000	M
2	Sharmilla	31	History	2008-03-24	20000	F
3	Sandeep	32	Mathematics	2016-12-12	30000	M
4	Sangeeta	35	History	2015-07-01	40000	F
5	Rakesh	42	Mathematics	2007-09-05	25000	M
6	Shyam	50	History	2008-06-27	30000	M
7	Shiv Om	44	Computer Sc	2017-02-25	21000	M
8	Shalakha	33	Mathematics	2018-07-31	20000	F

Table: Posting

P_ID	Department	Place
1	History	Agra
2	Mathematics	Raipur
3	Computer Science	Delhi

(i) `SELECT Department, COUNT(*) FROM Teacher GROUP BY Department;`

Ans.

Department	Count(*)
History	3
Computer Sc	2
Mathematics	3

(ii) `SELECT MAX(Date_of_Join), MIN(Date_of_Join) FROM Teacher;`

Ans.

MAX(Date_of_Join)	MIN(Date_of_Join)
2018-07-31	2007-09-05

(iii) `SELECT Teacher.Name, Teacher.Department, Posting.Place FROM Teacher, Posting WHERE Teacher.Department = Posting.Department AND Posting.Place="Delhi";`

Ans.

Name	Department	Place
Jugal	Computer Sc	Delhi
Shiv Om	Computer Sc	Delhi

(iv) To show all information about the teachers of History department.

Ans. `SELECT * FROM Teacher WHERE Department= "History";`

(v) To list the names of female teachers who are in Mathematics department.

Ans. `SELECT Name FROM Teacher WHERE Department= "Mathematics" AND Gender= "F";`

(vi) To list the names of all teachers with their date of joining in ascending order.

Ans. `SELECT Name FROM Teacher ORDER BY Date_of_join;`

(vii) To display teacher's name, salary, age for male teachers only.

Ans. `SELECT Name, Salary, Age FROM Teacher WHERE Gender='M';`

(viii) To display name, bonus for each teacher where bonus is 10% of salary.

Ans. `SELECT Name, Salary*0.1 AS Bonus FROM Teacher;`

19. Consider the following tables: COMPANY and MODEL.

Table: Company

Comp_ID	CompName	CompHO	ContactPerson
1	Titan	Okhla	C.B. Ajit
2	Ajanta	Najafgarh	R. Mehta
3	Maxima	Shahdara	B. Kohli
4	Seiko	Okhla	R. Chadha
5	Ricoh	Shahdara	J. Kishore

Note:

➤ Comp_ID is the Primary Key.

Table: Model

Model_ID	Comp_ID	Cost	DateOfManufacture
T020	1	2000	2010-05-12
M032	4	7000	2009-04-15
M059	2	800	2009-09-23
A167	3	1200	2011-01-12
T024	1	1300	2009-10-14

Note:

- Model_ID is the Primary Key.
- Comp_ID is the Foreign Key referencing Comp_ID of Company table.
- Write SQL commands for queries (i) to (iv) and output for (v) and (vi).
- (i) To display details of all models in the Model table in ascending order of DateOfManufacture.
- (ii) To display details of those models manufactured in 2011 and whose Cost is below 2000.
- (iii) To display the Model_ID, Comp_ID, Cost from the table Model, CompName and ContactPerson from Company table, with their corresponding Comp_ID.
- (iv) To decrease the cost of all the models in Model table by 15%.
- (v) `SELECT COUNT (DISTINCT CompHO) FROM Company;`
- (vi) `SELECT CompName, 'Mr. ', ContactPerson FROM Company WHERE CompName LIKE '%a';`

- Ans. (i) `SELECT * FROM Model
ORDER BY DateOfManufacture;`
- (ii) `SELECT * FROM Model
WHERE year (DateOfManufacture) = 2011 AND Cost < 2000;`
- (iii) `SELECT Model_ID, Comp_ID, Cost, CompName, ContactPerson FROM Model, Company
WHERE Model.Comp_ID = Company.Comp_ID;`
- (iv) `UPDATE Model
SET Cost = Cost - 0.15*Cost;`
- (v)

`count(distinct CompHO)`
3

(vi)

CompName	Mr.	ContactPerson
Ajanta	Mr.	R. Mehta
Maxima	Mr.	B. Kohli

20. Consider the following two tables: PRODUCT and CLIENT.

Table: Product

P_ID	ProductName	Manufacturer	Price	ExpiryDate
TP01	Talcum Powder	LAK	40	2011-06-26
FW05	Face Wash	ABC	45	2010-12-01
BS01	Bath Soap	ABC	55	2010-09-10
SH06	Shampoo	XYZ	120	2012-04-09
FW12	Face Wash	XYZ	95	2010-08-15

Note:

- P_ID is the Primary Key.

Table: Client

C_ID	ClientName	City	P_ID
1	Cosmetic Shop	Delhi	FW05
6	Total Health	Mumbai	BS01
12	Live Life	Delhi	SH06
15	Pretty One	Delhi	FW05
16	Dreams	Bengaluru	TP01
14	Expressions	Delhi	NULL

Note:

- C_ID is the Primary Key.
- P_ID is the Foreign Key referencing P_ID of Client table.

Write SQL statements for the queries (i) to (iv) and output for (v) and (vi):

- (i) To display the ClientName and City of all Mumbai- and Delhi-based clients in Client table.
- (ii) Increase the price of all the products in Product table by 10%.
- (iii) To display the ProductName, Manufacturer, ExpiryDate of all the products that expired on or before '2010-12-31'.
- (iv) To display C_ID, ClientName, City of all the clients (including the ones that have not purchased a product) and their corresponding ProductName sold.
- (v) `SELECT COUNT(DISTINCT Manufacturer) FROM Product;`
- (vi) `SELECT C_ID, Client_Name, City FROM Client WHERE City LIKE 'M%';`

Ans. (i) `SELECT ClientName, City FROM Client
WHERE City = 'Mumbai' OR City = 'Delhi';`

(ii) `UPDATE Product
SET Price = Price + 0.10 * Price;`

(iii) `SELECT ProductName, Manufacturer, ExpiryDate FROM Product
WHERE ExpiryDate <= '2010-12-31';`

(iv) `SELECT C_ID, ClientName, City, ProductName FROM Client LEFT JOIN Product
ON Client.P_ID = Product.P_ID;`

(v)

`COUNT(DISTINCT Manufacturer)`

3

(vi)

C_ID	Client_Name	City
6	Total Health	Mumbai

21. Consider the following two tables: Stationery and Consumer