Module – 5

Data Base

1. What do you understand By Database

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS).

2. What is Normalization?

Normalization is the process of organizing the data in the database.

Normalization is used to minimize the redundancy from a relation or set of relations. It is also used to eliminate undesirable characteristics like Insertion, Update, and Deletion Anomalies.

Normalization divides the larger table into smaller and links them using relationships.

The normal form is used to reduce redundancy from the database table.

3. What is Difference between DBMS and RDBMS?

DBMS	RDBMS
• This is high data redundancy. • Provides comparatively less security for data. • Difficult to modify data. • Takes more time to access data. • No keys and indexes. • Transactions are less secure and inefficient.	• This is low data redundancy. • Provides more security for data. • Easier to modify data. • Takes less time to access data. • Has keys and indexes. • Transactions are more secure and inefficient.

4. What is MF Cod Rule of RDBMS Systems?

Following are Codd's Twelve Principles of Relational Databases: Information is represented logically in tables. Data must be logically accessible by table, primary key, and column. Null values must be uniformly treated as “missing information,” not as empty strings, blanks, or zeros.

5. What do you understand By Data Redundancy?

Data redundancy is when multiple copies of the same information are stored in more than one place at a time. This challenge plagues organizations of all sizes in all industries and leads to elevated storage costs, errors, and compromised analytics.

6. What is DML Compiler in SQL?

A data manipulation language (DML) is a computer programming language used for adding (inserting), deleting, and modifying (updating) data in a database. A DML is often a sublanguage of a broader database language such as SQL, with the DML comprising some of the operators in the language.

7. What is SQL Key Constraints writing an Example of SQL Key Constraints

SQL constraints are used to specify rules for the data in a table. Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

8. What is save Point? How to create a save Point write a Query?

A SAVEPOINT is a point in a transaction in which you can roll the transaction back to a certain point without rolling back the entire transaction. Syntax for Savepoint command: SAVEPOINT SAVEPOINT_NAME; This command is used only in the creation of SAVEPOINT among all the transactions.

9. What is trigger and how to create a Trigger in SQL?

Creates a DML, DDL, or logon trigger. A trigger is a special type of stored procedure that automatically runs when an event occurs in the database server. DML triggers run when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view. These triggers fire when any valid event fires, whether table rows are affected or not. For more information, see DML Triggers.

Query :

Table : 1

1. Create Table Name : Student and Exam

- create table student(Rollno int auto_increment not null primary key,Name varchar(15) not null,Branch varchar(20) not null);
- insert into student (name,branch) values("jay","Computer science"),("suhani","Electronics and Com."),("Kirti","Electronics and Com");
- create table exam(Rollno int not null ,foreign key(Rollno) references student(Rollno),scode varchar(20),marks int(100),pcode varchar(20));

- insert into exam(Rollno,scode,marks,pcode) values(1,"cs11",50,"cs"),(1,"cs12",60,"cs"),(2,"ec101",66,"ec"),(2,"ec102",70,"ec"),(3,"ec101",45,"ec"),(3,"ec102",50,"ec");

Q2. Create table given below

- create table emp(FirstName varchar(20) not null,LastName varchar(20) not null,Address varchar(25) not null,City varchar(15) not null,Age int(5) not null);
- insert into emp(FirstName,LastName,Address,City,Age)
- values ("Mickey","Mouse","123 Fantasy Way","Anaheim",73),("Bat","Man","321 Cavern Ave","Gotham",54),("Wonder","Woman","987 Truth Way","Paradise",39),("Donald","Duck","555 Quack Street","Mallard",56),("Bugs","Bunny","567 Carrot Street","Rascal",58),("Wiley","Coyote","999 Acme Way","Canyon",61),("Cat","Woman","234 Purrfect Street","Hairball",32),("Tweety","Bird","543","Itotlow"
  ,28);

Q3.Create table Employee and Incentive.

Table Name: Employee

- create table Employee(Employee_id int auto_increment not null primary key,First_name varchar(15) not null,Last_name varchar(15) not null, Salary int(10) not null,Joining_date datetime not null,Department varchar(20) not null);
- insert into Employee(First_name,Last_name,Salary,Joining_date,Department)values('John','Abraham',10000000,'2013-01-01 12:00:00','Banking'),('Michael','Clarke',800000,'2013-01-01 12:00:00','Insurance'),('Roy','Thomes',700000,'2013-02-01 12:00:00','Banking'),('Tom','Jose',600000,'2013-02-01 12:00:00','Insurance'),('Jerry','Pinto',650000,'2013-02-01 12:00:00','Insurance'),('Philip','Mathew',750000,'2013-01-01 12:00:00','Service'),('TestName1','123',650000,'2013-01-01 12:00:00','Service'),('TestName2','Lname%',600000, '13-02-01 12:00:00','Insurance');

- create table Incentive(Employee_ref_id int not null,Incentive_date date not null,Incentive_amount int(10) not null);
- insert into Incentive (Employee_ref_id,Incentive_date,Incentive_amount)values(1,'2013-02-01',5000),(2,'2013-02-01',3000),(3,'2013-02-01',4000),(1,'2013-01-01',4500),(2,'2013-01-01',3500);

1. Get First_Name from employee table using Tom name “Employee Name”.
   - SELECT First_name FROM Employee WHERE First_name = 'Tom';
2. Get FIRST_NAME, Joining Date, and Salary from employee table.
   - SELECT First_name,Joining_date,Salary from Employee;

C. Get all employee details from the employee table order by First_Name Ascending and Salary descending?

- select * from Employee order by First_name asc,Salary desc;
  D.Get employee details from employee table whose first name contains ‘J’.
- select * from Employee where First_name like '%j%';

E. Get department wise maximum salary from

employee table order by salary ascending?
 SELECT Department, MAX(Salary) AS max_salary FROM employee GROUP BY Department ORDER BY max_salary ASC;

F. Select first_name, incentive amount from employeeand incentives table for those employees who have incentives and incentive amount greater than 3000.

- SELECT Employee.First_name,incentive.Incentive_amount FROM Employee JOIN incentive ON Employee.Employee_id=incentive.Employee_ref_id WHERE incentive.Incentive_amount > 3000;

G.Create After Insert trigger on Employee table which insert records in view table.

- create table viewtable(employee_id int auto_increment primary key ,first_name varchar(15) not null,last_name varchar(15) not null,salary int(10) not null,joining_date datetime not null,department varchar(20));
- create trigger emp_3_insert_trigger after insert on viewtable for each row insert into emp_3(first_name,last_name,salary,joining_date, department)values ('John','Abraham',10000000,'2013-1-01 12:00:00 ','Banking');

Q4. Create table given below: Salesperson and Customer.

- create table SALESPERSON(SNO int auto_increment not null primary key,SNAME varchar(15) not null,CITY varchar(20) not null,COMM float(5) not null);
- insert into SALESPERSON(SNO,SNAME,CITY,COMM)values (1001,'Peel','London',.12),(1002,'Serres','San Jose',.13),(1004,'Motika','London',.11),(1007,'Rafkin','Barcelona',.15),(1003,'Axelrod','New York',.1);
- create table CUSTOMER(CNM int(5) primary key not null ,CNAME varchar(15) not null,CITY varchar(15) not null,RATING int(5) not null,SNO int(10),foreign key(SNO) references SALESPERSON(SNO));
- insert into CUSTOMER(CNM,CNAME,CITY,RATING,SNO)values(201,'Hoffman','London',100,1001),(202,'Giovanne','Roe',200,1003),(203,'Liu','San Jose',300,1002),(204,'Grass','Barcelona',100,1002),(206,'Clements','London',300,1007),(207,'Pereria','Roe',100,1004);

❖ Retrieve the below data from above table

A)All orders for more than $1000.

- Select * from CUSTOMER where order >= 1000;

B) Names and cities of all salespeople in London with commission above 0.12.

- select SNAME,CITY,COMM from SALESPERSON where (CITY='London' and COMM>'.12');

C) All salespeople either in Barcelona or in London

- SELECT * FROM salesperson WHERE (CITY = 'Barcelona' OR CITY = 'London');

D)All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded).

- SELECT * FROM salesperson WHERE COMM > 0.10 AND COMM < 0.12;

E) All customers excluding those with rating <= 100 unless they are located in Rome.

- SELECT *FROM CUSTOMER WHERE NOT (RATING <= 100 AND CITY = 'Rome');