

Module-5

1) What do you understand By Database

A database is an electronically stored, systematic collection of data.

2) What is Normalization?

Normalization is the process of minimizing redundancy from a relation or set of relations. Redundancy in relation may cause insertion, deletion, and update anomalies. So, it helps to minimize redundancies in relations.

3) What is Difference between DBMS and RDBMS?

DBMS:

- Data stored is in the file format.
- Individual access to data elements.
- Data stored is a small quantity.
- DBMS supports a single user.
- The software and hardware requirements are low.

Example of dbms :- XML, Microsoft access.

RDBMS:

- Data stored is in table format.
- Multiple data elements are accessible together.
- Data is stored in a large amount.
- RDBMS supports multiple users.
- The software and hardware requirements are higher.

Example of rdms :- Oracle, SQL Server.

4) What do you understand By Data Redundancy?

Codd's Rules are a set of principles defined by Dr. Edgar F. Codd, the father of the relational database model. These rules outline the essential characteristics and requirements for a database management system (DBMS) to be considered a true relational database management system (RDBMS).

- **12 Codd's rule in RDBMS:** The Foundation Rule, Information Rule, Guaranteed Access Rule, Systematic Treatment of Null Values, Active Online Catalog, Comprehensive Data Sublanguage, view updating,

Relational Level Operation, Physical Data independence, Logical Data independence, integrity independence,

Distribution independence, Non-Subversion.

5) What do you understand By Data Redundancy?

Data redundancy occurs when the same piece of data exists in multiple places, whereas data inconsistency is when the same data exists in different formats in multiple tables. Unfortunately, data redundancy can cause data inconsistency, which can provide a company with unreliable and/or meaningless information.

6) What is a DDL Interpreter?

DDL commands are used to define and manage database structures such as tables and indexes. Common DDL commands include CREATE, ALTER, DROP, and TRUNCATE.

- 'CREATE' TABLE will create a new table in the database.

- 'ALTER' TABLE will modify the structure of an existing table.

- 'DROP' TABLE will remove a table from the database.
- 'TRUNCATE' TABLE will remove all rows from a table, quickly clearing its contents without logging individual row deletions.

7) What is DML Compiler in SQL?

DML commands are used to interact with and modify the data within a database. Common DML commands include :
insert,update,delete,call,explaincall

- INSERT: Adds new data into the database.

UPDATE: Modifies existing data in the database.

DELETE: Removes data from the database.

8) What is SQL Key Constraints writing an Example of SQL Key Constraints?

SQL key constraints are rules that you apply to database columns to ensure the integrity and accuracy of the data. These constraints help control what kind of data can be stored in a column and how it relates to other data within the database.

NOT NULL - Ensures that a column cannot have a NULL value.

Example:

- CREATE TABLE Employees (EmployeeID INT PRIMARY KEY, Name VARCHAR (100) NOT NULL, Position VARCHAR (100));

UNIQUE - Ensures that all values in a column are different.

Example:

- CREATE TABLE Teachers (TeacherID INT PRIMARY KEY, Email VARCHAR (100) UNIQUE, Name VARCHAR (100));

PRIMARY KEY - A combination of NOT NULL and UNIQUE. Uniquely identifies each row in a table.

Example:

- CREATE TABLE Students (StudentID INT PRIMARY KEY, Name VARCHAR (100), Age INT);

FOREIGN KEY - Prevents actions that would destroy links between tables.

Example:

- CREATE TABLE Enrollments (EnrollmentID INT PRIMARY KEY, StudentID INT, CourseID INT, FOREIGN KEY (StudentID) REFERENCES Students(StudentID));

CHECK - Ensures that the values in a column satisfies a specific condition

Example :

- CREATE TABLE Products (ProductID INT PRIMARY KEY, ProductName VARCHAR(100), Price DECIMAL(10, 2), CHECK (Price > 0));

DEFAULT - Sets a default value for a column if no value is specified.

Example :

- CREATE TABLE Orders (OrderID INT PRIMARY KEY, OrderDate DATE DEFAULT GETDATE(), Quantity INT DEFAULT 1);

CREATE INDEX - Used to create and retrieve data from the database very quickly.

Example :

- CREATE INDEX idx_LastName ON Employees(LastName);

9) What is save Point? How to create a save Point write a Query?

A savepoint is like a bookmark for your database changes. You can set a savepoint before making some changes. If something goes wrong, you can go back to that savepoint and undo all the changes made after that point. It's a way to protect your database from accidental errors.

- `SAVEPOINT first_savepoint;`

10) What is 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.

1)

```
CREATE TABLE student
```

```
(  
    rollno int PRIMARY KEY,  
    name varchar(50),  
    branch varchar(50)  
);
```

```
CREATE TABLE exam
```

```
(  
    rollno int,  
    s_code varchar(50),  
    marks float(20),  
    p_code varchar(50),  
  
    FOREIGN KEY(rollno) REFERENCES student(rollno)  
);
```

```
INSERT INTO student VALUES(1,"jay","computer science");
```

```
INSERT INTO student VALUES(2,"suhani","electronic and com");
```

```
INSERT INTO student VALUES(3,"kriti","electronic and com");
```

```
INSERT INTO exam VALUES(1,'cs11',50,'cs');
```

```
INSERT INTO exam VALUES(1,'cs12',60,'cs');
```

```
INSERT INTO exam VALUES(2,'ec101',66,'ec');
```

```
INSERT INTO exam VALUES(2,'ec102',70,'ec');
```

```
INSERT INTO exam VALUES(3,'ec101',45,'ec');
```

```
INSERT INTO exam VALUES(3,'ec102',50,'ec');
```

	rollno	name	branch
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	jay	computer science
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	suhani	electronic and com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	kriti	electronic and com

rollno	s_code	marks	p_code
1	cs11	50	cs
1	cs12	60	cs
2	ec101	66	ec
2	ec102	70	ec
3	ec101	45	ec
3	ec102	50	ec

2)

CREATE TABLE employee

(
 employee_id int,
 first_name varchar(50),
 last_name varchar(50),
 salary int,
 joining_date date,
 department varchar(50)
);

INSERT into employee VALUES(1,"john","abraham",1000000,'2001-01-13',"banking");
 INSERT into employee VALUES(2,"michael","clarke",800000,'2001-01-13',"insurance");
 INSERT into employee VALUES(3,"roy","thomas",700000,'2001-02-13',"banking");
 INSERT into employee VALUES(4,"tom","jose",600000,'2001-02-13',"insurance");
 INSERT into employee VALUES(5,"jerry","pinto",650000,'2001-02-13',"insurance");

```

INSERT into employee VALUES(6,"philip","mathew",750000,'2001-01-13',"services");
INSERT into employee VALUES(7,"testname1","123",650000,'2001-01-13',"services");
INSERT into employee VALUES(8,"testname2","lname%",600000,'2001-02-13',"insurance");

```

employee_id	first_name	last_name	salary	joining_date	department
1	john	abraham	1000000	2001-01-13	banking
2	michael	clarke	800000	2001-01-13	insurance
3	roy	thomas	700000	2001-02-13	banking
4	tom	jose	600000	2001-02-13	insurance
5	jerry	pinto	650000	2001-02-13	insurance
6	philip	mathew	750000	2001-01-13	services
7	testname1	123	650000	2001-01-13	services
8	testname2	lname%	600000	2001-02-13	insurance

```

CREATE TABLE incentive

```

```

(
    employee_ref_id int,
    incentive_date date,
    incentive_amount int
);

```

```

INSERT INTO incentive VALUES(1,'2001-02-13',5000);
INSERT INTO incentive VALUES(2,'2001-02-13',3000);
INSERT INTO incentive VALUES(3,'2001-02-13',4000);
INSERT INTO incentive VALUES(1,'2001-01-13',4500);
INSERT INTO incentive VALUES(2,'2001-01-13',3500);

```

employee_ref_id	incentive_date	incentive_amount
1	2001-02-13	5000
2	2001-02-13	3000
3	2001-02-13	4000
1	2001-01-13	4500
2	2001-01-13	3500

3)

- SELECT * from employee WHERE first_name="tom";

employee_id	first_name	last_name	salary	joining_date	department
4	tom	jose	600000	2001-02-13	insurance

4)

- SELECT first_name,joining_date,salary from employee;

first_name	joining_date	salary
john	2001-01-13	1000000
michael	2001-01-13	800000
roy	2001-02-13	700000
tom	2001-02-13	600000
jerry	2001-02-13	650000
philip	2001-01-13	750000
testname1	2001-01-13	650000
testname2	2001-02-13	600000

5)

- SELECT * FROM employee ORDER BY first_name ASC, salary DESC;

employee_id	first_name ▲ 1	last_name	salary ▼ 2	joining_date	department
5	jerry	pinto	650000	2001-02-13	insurance
1	john	abraham	1000000	2001-01-13	banking
2	michael	clarke	800000	2001-01-13	insurance
6	philip	mathew	750000	2001-01-13	services
3	roy	thomas	700000	2001-02-13	banking
7	testname1	123	650000	2001-01-13	services
8	testname2	lname%	600000	2001-02-13	insurance
4	tom	jose	600000	2001-02-13	insurance

6)

- SELECT * FROM employee WHERE first_name LIKE 'j%';

employee_id	first_name	last_name	salary	joining_date	department
1	john	abraham	1000000	2001-01-13	banking
5	jerry	pinto	650000	2001-02-13	insurance

7)

- SELECT Department, MAX(Salary) AS Max_Salary FROM employee GROUP BY Department ORDER BY Max_Salary ASC;

Department	Max_Salary
services	750000
insurance	800000
banking	1000000

9)

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

First_name	Incentive_amount
john	5000
john	4500
michael	3500
roy	4000

10)

CREATE TABLE ViewHistory (

```

Employee_id INT,
First_name VARCHAR (20),
Last_name VARCHAR (20),
Salary INT,
Joining_date DATETIME,
Department VARCHAR (20),
date_time timestamp,
action_performed text
);
- DELIMITER $$

CREATE TRIGGER AfterEmployeeInsert AFTER INSERT ON Employee FOR EACH ROW
BEGIN
INSERT INTO ViewHistory (Employee_id, First_name, Last_name, Salary, Joining_date,
Department ,action_performed)
VALUES (NEW.Employee_id, NEW.First_name, NEW.Last_name, NEW.Salary, NEW.Joining_date,
NEW.Department,'Record inserted');
END;
- INSERT INTO Employee VALUES (9, 'Alice', 'Williams', 900000, '2024-08-08 09:00:00', 'IT');

```

Employee_id	First_name	Last_name	Salary	Joining_date	Department	date_time	action_performed
9	Alice	Williams	900000	2024-08-08 09:00:00	IT	2024-08-08 22:30:10	Record inserted

11)

```

CREATE TABLE salesperson
(
SNo int PRIMARY KEY,
SNAME varchar(50),
CITY varchar(50),
COMM decimal(5,2)
);

```

```
INSERT INTO Salesperson VALUES(1001,"Peel","London",.12);
INSERT INTO Salesperson VALUES(1002,"Serres","San Jose",.13);
INSERT INTO Salesperson VALUES(1004,"Motika","London",.11);
INSERT INTO Salesperson VALUES(1007,"Rafkin","Barcelona",.15);
INSERT INTO Salesperson VALUES(1003,"Axelrod","New York",.1);
```

```
CREATE TABLE Customer
```

```
(
    CNM int,
    CNAME varchar(50),
    CITY varchar(50),
    RATING int,
    SNo int,

    PRIMARY KEY(CNM),
    FOREIGN KEY(SNo) REFERENCES salesperson (SNo)
);
```

```
INSERT INTO Customer VALUES(201, 'Hoffman', 'London', 100,1001);
INSERT INTO Customer VALUES(202, 'Giovanne', 'Roe', 200, 1003);
INSERT INTO Customer VALUES(203, 'Liu', 'San Jose', 300, 1002);
INSERT INTO Customer VALUES(204, 'Grass', 'Barcelona', 200, 1007);
INSERT INTO Customer VALUES(206, 'Clemens', 'London', 300, 1001);
INSERT INTO Customer VALUES(207, 'Pereira', 'Roe', 100, 1004);
```

CNM	CNAME	CITY	RATING	SNo
201	Hoffman	London	100	1001
202	Giovanne	Roe	200	1003
203	Liu	San Jose	300	1002
204	Grass	Barcelona	200	1007
206	Clemens	London	300	1001
207	Pereira	Roe	100	1004

SNo	SNAME	CITY	COMM
1001	Peel	London	0.12
1002	Serres	San Jose	0.13
1003	Axelrod	New York	0.10
1004	Motika	London	0.11
1007	Rafkin	Barcelona	0.15

13)

- SELECT * FROM customer WHERE rating > 100;

<div><div><div></div><div></div><div></div></div></div>				CNM	CNAME	CITY	RATING	SNo
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	202	Giovanne	Roe	200	1003
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	203	Liu	San Jose	300	1002
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	204	Grass	Barcelona	200	1007
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	206	Clemens	London	300	1001











14)

- SELECT sname,City FROM salesperson WHERE City = 'london' AND comm > 0.12;

sname	City
-------	------

15)

- SELECT * FROM salesperson WHERE City IN ('Barcelona' ,'London');

				SNo	SNAME	CITY	COMM
<input type="checkbox"/>		Edit		Copy		Delete	1001 Peel London 0.12
<input type="checkbox"/>		Edit		Copy		Delete	1004 Motika London 0.11
<input type="checkbox"/>		Edit		Copy		Delete	1007 Rafkin Barcelona 0.15








16)

- SELECT * FROM Salesperson WHERE Comm > 0.10 AND Comm < 0.12;

				SNo	SNAME	CITY	COMM
<input type="checkbox"/>		Edit		Copy		Delete	1004 Motika London 0.11

17)

- SELECT * FROM customer WHERE RATING<=100;

				CNM	CNAME	CITY	RATING	SNo
<input type="checkbox"/>		Edit		Copy		Delete	201 Hoffman London 100	1001
<input type="checkbox"/>		Edit		Copy		Delete	207 Pereira Roe 100	1004

18)

CREATE TABLE salesman

(

Salesman_id INT,
Name varchar (50),
City varchar (50),
Commission float(20)

);

INSERT INTO salesman VALUES (5001,'James Hoog','New York', 0.15);

INSERT INTO salesman VALUES (5002,'Nail Knite', 'Paris', 0.13);

```

INSERT INTO salesman VALUES (5005,'Pit Alex', 'London', 0.11);
INSERT INTO salesman VALUES (5006,'Mc Lyon', 'Paris', 0.14);
INSERT INTO salesman VALUES (5007,'Paul Adam', 'Rome', 0.13);
INSERT INTO salesman VALUES (5003,'Lauson Hen', 'San Jose', 0.12);

```

Salesman_id	Name	City	Commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

19)

```
CREATE TABLE orders
```

```

(
    ord_no INT,
    purch_amt float(20),
    ord_date DATE,
    customer_id INT,
    salesman_id INT
);

```

```

INSERT INTO orders VALUES (70001, 150.50, '2012-10-05', 3005, 5002);
INSERT INTO orders VALUES (70009, 270.65, '2012-09-10', 3001, 5005);
INSERT INTO orders VALUES (70002, 65.26, '2012-10-05', 3002, 5001);
INSERT INTO orders VALUES (70004, 110.50, '2012-08-17', 3009, 5003);
INSERT INTO orders VALUES (70007, 948.50, '2012-09-10', 3005, 5002);
INSERT INTO orders VALUES (70005, 2400.60, '2012-07-27', 3007, 5001);
INSERT INTO orders VALUES (70008, 5760.00, '2012-09-10', 3002, 5001);
INSERT INTO orders VALUES (70010, 1983.43, '2012-10-10', 3004, 5006);
INSERT INTO orders VALUES (70003, 2480.40, '2012-10-10', 3009, 5003);
INSERT INTO orders VALUES (70012, 250.45, '2012-06-27', 3008, 5002);

```

INSERT INTO orders VALUES (70011, 75.29, '2012-08-17', 3003, 5007);

INSERT INTO orders VALUES (70013, 3045.60, '2012-04-25', 3002, 5001);

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70007	948.5	2012-09-10	3005	5002
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001

- SELECT ord_no, ord_date, purch_amt FROM orders WHERE salesman_id = 5001;

ord_no	ord_date	purch_amt
70002	2012-10-05	65.26
70005	2012-07-27	2400.6
70008	2012-09-10	5760
70013	2012-04-25	3045.6

20)

CREATE TABLE item_mast

(

 pro_id INT PRIMARY KEY,
 pro_name VARCHAR (50),
 pro_price float (20),
 pro_com INT);































INSERT INTO item_mast VALUES (101,'Mother Board' , 3200.00,15);

INSERT INTO item_mast VALUES (102, 'Key Board', 450.00, 16);

















```

INSERT INTO item_mast VALUES (103, 'ZIP drive', 250.00, 14);
INSERT INTO item_mast VALUES (104, 'Speaker', 550.00, 16);
INSERT INTO item_mast VALUES (105, 'Monitor', 5000.00, 11);
INSERT INTO item_mast VALUES (106, 'DVD drive', 900.00, 12);
INSERT INTO item_mast VALUES (107, 'CD drive', 800.00, 12);
INSERT INTO item_mast VALUES (108, 'Printer', 2600.00, 13);
INSERT INTO item_mast VALUES (109, 'Refill cartridge', 350.00, 13);
INSERT INTO item_mast VALUES (110, 'Mouse', 250.00, 12);

```

				pro_id	pro_name	pro_price	pro_com
<input type="checkbox"/>				101	Mother Board	3200	15
<input type="checkbox"/>				102	Key Board	450	16
<input type="checkbox"/>				103	ZIP drive	250	14
<input type="checkbox"/>				104	Speaker	550	16
<input type="checkbox"/>				105	Monitor	5000	11
<input type="checkbox"/>				106	DVD drive	900	12
<input type="checkbox"/>				107	CD drive	800	12
<input type="checkbox"/>				108	Printer	2600	13
<input type="checkbox"/>				109	Refill cartridge	350	13
<input type="checkbox"/>				110	Mouse	250	12

- SELECT pro_id,pro_name,pro_price,pro_com FROM item_mast WHERE PRO_PRICE BETWEEN 200 AND 600;

				pro_id	pro_name	pro_price	pro_com
<input type="checkbox"/>				102	Key Board	450	16
<input type="checkbox"/>				103	ZIP drive	250	14
<input type="checkbox"/>				104	Speaker	550	16
<input type="checkbox"/>				109	Refill cartridge	350	13
<input type="checkbox"/>				110	Mouse	250	12































21)

- SELECT AVG(PRO_PRICE) avg_price FROM item_mast WHERE PRO_COM = 16;

avg_price
500
































22)

- SELECT pro_name AS "Item_Name",pro_price AS "Price in RS." FROM item_mast ;

<div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div></div><div></div></div>						Item_Name	Price in RS.	
<input type="checkbox"/>		Edit		Copy		Delete	Mother Board	3200
<input type="checkbox"/>		Edit		Copy		Delete	Key Board	450
<input type="checkbox"/>		Edit		Copy		Delete	ZIP drive	250
<input type="checkbox"/>		Edit		Copy		Delete	Speaker	550
<input type="checkbox"/>		Edit		Copy		Delete	Monitor	5000
<input type="checkbox"/>		Edit		Copy		Delete	DVD drive	900
<input type="checkbox"/>		Edit		Copy		Delete	CD drive	800
<input type="checkbox"/>		Edit		Copy		Delete	Printer	2600
<input type="checkbox"/>		Edit		Copy		Delete	Refill cartridge	350
<input type="checkbox"/>		Edit		Copy		Delete	Mouse	250





23)

- SELECT pro_name , pro_price FROM item_mast WHERE pro_price >= 250 ORDER BY pro_price DESC, pro_name ASC;

		pro_name ▲ 2	pro_price ▼ 1
<input type="checkbox"/>	 Edit  Copy  Delete	Monitor	5000
<input type="checkbox"/>	 Edit  Copy  Delete	Mother Board	3200
<input type="checkbox"/>	 Edit  Copy  Delete	Printer	2600
<input type="checkbox"/>	 Edit  Copy  Delete	DVD drive	900
<input type="checkbox"/>	 Edit  Copy  Delete	CD drive	800
<input type="checkbox"/>	 Edit  Copy  Delete	Speaker	550
<input type="checkbox"/>	 Edit  Copy  Delete	Key Board	450
<input type="checkbox"/>	 Edit  Copy  Delete	Refill cartridge	350
<input type="checkbox"/>	 Edit  Copy  Delete	Mouse	250
<input type="checkbox"/>	 Edit  Copy  Delete	ZIP drive	250

24)

- SELECT AVG(PRO_PRICE) AS average_price , PRO_COM AS companycode FROM item_mast GROUP BY PRO_COM;

		average_price	companycode
<input type="checkbox"/>	 Edit  Copy  Delete	5000	11
<input type="checkbox"/>	 Edit  Copy  Delete	650	12
<input type="checkbox"/>	 Edit  Copy  Delete	1475	13
<input type="checkbox"/>	 Edit  Copy  Delete	250	14
<input type="checkbox"/>	 Edit  Copy  Delete	3200	15
<input type="checkbox"/>	 Edit  Copy  Delete	500	16