**MODULE: 5 (DataBase)**

=================================================================================

1. **Create two tables Student and Exam and link two tables through Primary Key and Foreign Key.**

=================================================================================

CREATE TABLE Student

(

Rollno int PRIMARY KEY,

Name varchar(20),

Branch varchar(20)

)

INSERT INTO student(Rollno,Name,Branch) VALUES(1,"Jay","Computer Science");

INSERT INTO student(Rollno,Name,Branch) VALUES(2,"Suhani","Electronic and Com");

INSERT INTO student(Rollno,Name,Branch) VALUES(3,"Kriti","Electronic and Com");

CREATE TABLE Exam

(

Rollno int,

S\_code varchar(20),

Marks int,

P\_code varchar(20),

FOREIGN KEY(Rollno) REFERENCES student(Rollno)

)

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(1,"CS11",50,"CS");

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(1,"CS12",60,"CS");

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(2,"EC101",66,"EC");

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(2,"EC102",70,"EC");

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(3,"EC101",45,"EC");

INSERT INTO exam(Rollno,S\_code,Marks,P\_code) VALUES(3,"EC102",50,"EC");

======================================================================================

**----> Create two tables Employee and Incentive and link two tables**

======================================================================================

CREATE TABLE Employee

(

Employee\_id int PRIMARY KEY,

First\_name varchar(20),

Last\_name varchar(20),

Salary int,

Joining\_date date,

Department varchar(20)

)

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(1,"John","Abraham",1000000,"01-JAN-13 12.00.00 AM","Banking");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(2,"Michael","Clarke",800000,"01-JAN-13 12.00.00 AM","Insurance");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(3,"Roy","Thomas",700000,"01-FEB-13 12.00.00 AM","Banking");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(4,"Tom","Jose",600000,"01-FEB-13 12.00.00 AM","Insurance");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(5,"Jerry","Pinto",650000,"01-FEB-13 12.00.00 AM","Insurance");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(6,"Philip","Mathew",750000,"01-JAN-13 12.00.00 AM","Services");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(7,"TestName1","123",650000,"01-JAN-13 12.00.00 AM","Services");

INSERT INTO employee(Employee\_id,First\_name,Last\_name,Salary,Joining\_date,Department)

VALUES(8,"TestName2","Lname%",600000,"01-FEB-13 12.00.00 AM","Insurance");

CREATE TABLE Incentive

(

Employee\_ref\_id int,

Incentive\_date date,

Incentive\_amount int,

FOREIGN KEY(Employee\_ref\_id) REFERENCES employee(Employee\_id)

)

INSERT INTO incentive(Employee\_ref\_id,Incentive\_date,Incentive\_amount) VALUES(1,"01-FEB-13",5000);

INSERT INTO incentive(Employee\_ref\_id,Incentive\_date,Incentive\_amount) VALUES(2,"01-FEB-13",3000);

INSERT INTO incentive(Employee\_ref\_id,Incentive\_date,Incentive\_amount) VALUES(3,"01-FEB-13",4000);

INSERT INTO incentive(Employee\_ref\_id,Incentive\_date,Incentive\_amount) VALUES(1,"01-JAN-13",4500);

INSERT INTO incentive(Employee\_ref\_id,Incentive\_date,Incentive\_amount) VALUES(2,"01-JAN-13",3500);

1. **Get First\_Name from employee table using Tom name “Employee Name”.**

SELECT \* FROM employee WHERE First\_name="Tom"

1. **Get FIRST\_NAME, Joining Date, and Salary from employee table.**

SELECT First\_name, Joining\_date, Salary FROM employee

1. **Get all employee details from the employee table order by First\_Name Ascending and Salary**

**descending?**

SELECT \* FROM employee ORDER BY First\_name ASC

SELECT \* FROM employee ORDER BY Salary DESC

1. **Get employee details from employee table whose first name contains ‘J’.**

SELECT \* FROM employee WHERE First\_name LIKE 'J%'

1. **Get department wise maximum salary from employee table order by salary ascending?**

SELECT \* FROM employee ORDER BY Salary ASC

SELECT MAX(Salary), Department FROM employee WHERE Department="Banking"

SELECT MAX(Salary), Department FROM employee WHERE Department="Insurance"

SELECT MAX(Salary), Department FROM employee WHERE Department="Services"

1. **Select first\_name, incentive amount from employee and incentives table forthose employees who**

**have incentives and incentive amount greater than 3000**

SELECT employee.First\_name, incentive.Incentive\_amount FROM employee

INNER JOIN incentive ON employee.Employee\_id=incentive.Employee\_ref\_id

SELECT \* FROM incentive WHERE Incentive\_amount>3000

1. **Create After Insert trigger on Employee table which insert records in viewtable.**

CREATE TABLE viewtable

(

id int,

fname varchar(20),

lname varchar(20),

salary int,

jdate date,

department varchar(20),

date\_time timestamp,

action\_performed varchar(40)

)

=================================== **TRIGGER START** ==================================

DELIMITER $$

CREATE TRIGGER insert\_trigger AFTER INSERT ON employee FOR EACH ROW

BEGIN

INSERT INTO viewtable(id, fname, lname, salary, jdate, department, action\_performed)

VALUES(new.Employee\_id, new.First\_name, new.Last\_name, new.Salary, new.Joining\_date, new.Department, "Record Inserted!");

END

==================================== **TRIGGER END** ===================================

===================================================================================

**----> Create table given below: Salesperson and Customer**

===================================================================================

CREATE TABLE Salesperson

(

SNo int PRIMARY KEY,

SNAME varchar(20),

CITY varchar(20),

COMM float

)

INSERT INTO salesperson(SNo,SNAME,CITY,COMM) VALUES(1001,"Peel","London",0.12);

INSERT INTO salesperson(SNo,SNAME,CITY,COMM) VALUES(1002,"Serres","San Jose",0.13);

INSERT INTO salesperson(SNo,SNAME,CITY,COMM) VALUES(1004,"Motika","London",0.11);

INSERT INTO salesperson(SNo,SNAME,CITY,COMM) VALUES(1007,"Rafkin","Barcelona",0.15);

INSERT INTO salesperson(SNo,SNAME,CITY,COMM) VALUES(1003,"Axelrod","New York",0.1);

CREATE TABLE Customer

(

CNM int PRIMARY KEY,

CNAME varchar(20),

CITY varchar(20),

RATING int,

SNo int,

FOREIGN KEY(SNo) REFERENCES salesperson(SNo)

)

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(201,"Hoffman","London",100,1001);

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(202,"Giovanne","Roe",200,1003);

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(203,"Liu","San Jose",300,1002);

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(204,"Grass","Barcelona",100,1002);

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(206,"Ciemens","London",300,1007);

INSERT INTO customer(CNM,CNAME,CITY,RATING,SNo) VALUES(207,"Pereira","Roe",100,1004);

1. **Names and cities of all salespeople in London with commission above 0.12**

SELECT SANAME,CITY FROM salesperson WHERE CITY='London' AND COMM > 0.12;

1. **All salespeople either in Barcelona or in London**

SELECT SNAME FROM salesperson WHERE CITY="London" OR "Barcelona"

1. **All salespeople with commission between 0.10 and 0.12. (Boundary valuesshould be excluded).**

SELECT SNAME,COMM FROM salesperson WHERE COMM BETWEEN 0.10 AND 0.12

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

SELECT \* FROM customers WHERE RATING>100 OR CITY='Rome';

1. **Write a SQL statement that displays all the information about all salespeople**

SELECT \* FROM salesperson

===================================================================================

**-----> Create table given below: salesman and orders**

===================================================================================

CREATE TABLE salesman

(

salesman\_id int PRIMARY KEY,

name varchar(20),

city varchar(20),

commision float

)

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5001,"James Hoog","New York",0.15);

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5002,"Nail Knite","Paris",0.13);

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5005,"Pit Alex","London",0.11);

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5006,"Mc Lyon","Paris",0.14);

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5007,"Paul Adam","Rome",0.13);

INSERT INTO salesman(salesman\_id,name,city,commision) VALUES(5003,"Lauson Hen","San Jose",0.12);

CREATE TABLE orders

(

ord\_no int PRIMARY KEY,

purch\_amt int,

ord\_date date,

customer\_id int,

salesman\_id int,

FOREIGN KEY(salesman\_id) REFERENCES salesman(salesman\_id)

)

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70001,150.5,"2012-10-05",3005,5002);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70009,270.65,"2012-09-10",3001,5005);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70002,65.26,"2012-10-05",3002,5001);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70004,110.5,"2012-08-17",3009,5003);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70007,948.5,"2012-09-10",3005,5002);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70005,2400.6,"2012-07-27",3007,5001);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70008,5760,"2012-09-10",3002,5001);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70010,1983.43,"2012-10-10",3004,5006);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70003,2480.4,"2012-10-10",3009,5003);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70012,250.45,"2012-06-27",3008,5002);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70011,75.29,"2012-07-17",3003,5007);

INSERT INTO orders(ord\_no,purch\_amt,ord\_date,customer\_id,salesman\_id)

VALUES(70013,3045.6,"2012-04-25",3002,5001);

1. **All orders for more than $1000.**

SELECT \* FROM orders WHERE purch\_amt>1000;

1. **From the following table, write a SQL query to find orders that are delivered by a salesperson with**

**ID. 5001. Return ord\_no, ord\_date, purch\_amt.**

SELECT ord\_no,purch\_amt,ord\_date FROM orders WHERE salesman\_id=5001

===========================================================================

**----> Create table item\_mast**

===========================================================================

CREATE TABLE item\_mast

(

PRO\_ID int PRIMARY KEY,

PRO\_NAME varchar(40),

PRO\_PRICE float,

PRO\_COM int

)

============================== **PROCEDURE START** ====================================

DELIMITER $$

CREATE PROCEDURE insert\_data(i int, j varchar(40), k float, l int)

BEGIN

INSERT INTO item\_mast(PRO\_ID, PRO\_NAME, PRO\_PRICE, PRO\_COM) VALUES(i,j,k,l);

END

CALL insert\_data(101,"Mother Board",3200.00,15);

CALL insert\_data(102,"Key Board",450.00,16);

CALL insert\_data(103,"Zip Drive",250.00,14);

CALL insert\_data(104,"Speaker",550.00,16);

CALL insert\_data(105,"Monitor",5000.00,11);

CALL insert\_data(106,"DVD drive",900.00,12);

CALL insert\_data(107,"CD drive",800.00,12);

CALL insert\_data(108,"Printer",2600.00,13);

CALL insert\_data(109,"Refill Cartridge",350.00,13);

CALL insert\_data(110,"Mouse",250.00,12);

============================== **PROCEDURE END** ====================================

1. **From the following table, write a SQL query to select a range of products whose price is in the range**

**Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and**

**pro\_com.**

SELECT \* FROM item\_mast WHERE PRO\_PRICE BETWEEN 200 AND 600

1. **From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg.**

SELECT AVG(PRO\_PRICE) FROM item\_mast WHERE PRO\_COM=16

1. **From the following table, write a SQL query to display the pro\_name as 'Item Name' and pro\_priceas 'Price in Rs.'**

SELECT PRO\_NAME AS Item\_Name, PRO\_PRICE AS Price\_in\_Rs FROM item\_mast

1. **From the following table, write a SQL query to find the items whose prices are higher than or equal**

**to $250. Order the result by product price in descending, then product name in ascending. Return**

**pro\_name and pro\_price.**

SELECT PRO\_NAME,PRO\_PRICE FROM item\_mast WHERE PRO\_PRICE>=250

SELECT PRO\_NAME FROM item\_mast ORDER BY PRO\_NAME ASC

SELECT PRO\_PRICE FROM item\_mast ORDER BY PRO\_PRICE DESC

1. **From the following table, write a SQL query to calculate average price of the items for each**

**company. Return average price and company code.**

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=11

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=12

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=13

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=14

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=15

SELECT AVG(PRO\_PRICE), PRO\_COM FROM item\_mast WHERE PRO\_COM=16