1. Create Table Name : Student and Exam

# Student table:-

CREATE TABLE student(ROllno int,Name char(20),Branch char(30)); ALTER table student1 add PRIMARY KEY(ROllno);

# Insert data:-

INSERT into student VALUES

(1,jay,computer science),

(2,suhani,electronic and com),

(3,kriti,electronic and com);

# Exam table:-

CREATE table exam(ROllno int,S\_code varchar(20),Marks int,P\_code varchar(20));

# Add Foreign key:-

ALTER TABLE exam add FOREIGN key (Rollno) REFERENCES exam (ROllno);

# Insert data:-

INSERT INTO exam 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 );

2. Create table given below: Employee and IncentiveTable **Create table employee:-**

CREATE table employee (employee\_id int PRIMARY KEY AUTO\_INCREMENT,first\_name char(20),last\_name varchar(20),salary int,joining\_date datetime,deparment char(30));

**Insert data:-**

INSERT INTO employee ( first\_name , last\_name , salary ,

joining\_date , deparment ) VALUES

( john , abraham ,1000000, 2013-01-01 12:00:00 , Banking ),

( michael , clarke ,800000, 2013-01-01 12:00:00 , Insurance ),

( Roy , Thomas ,700000, 2013-02-01 12:00:00 , Banking ),

( Tom , Jose ,600000, 2013-02-01 12:00:00 , Insurance ),

( Jerry , pinro ,650000, 2013-02-01 12:00:00 , Insurance ), ( Philip , Mathew ,750000, 2013-01-01 12:00:00 , Services ),

( TestName1 , 123 ,650000, 2013-01-01 12:00:00 , Services ),

( TestName2 , Lname% ,600000, 2013-02-01 12:00:00 , Insuance ); **Create table incentive:-**

CREATE TABLE Incentive (employee\_ref\_id int,Incentive\_dare datetime,Incentive\_amount int);

## Insert data:-

INSERT INTO incentive VALUES (1, 2013-02-01 , 5000 ), (2, 2013-0201 ,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

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

SELECT \* FROM employee ORDER by first\_name asc ,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 salaryascending?

SELECT MAX(salary) from employee ORDER BY salary AS;

1. Select first\_name, incentive amount from employee and incentivestable forthose employees who have incentives and incentive amount greater than 3000

SELECT first\_name from employee WHERE salary>3000;

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

DELIMITER $$

CREATE TRIGGER em AFTER INSERT on employee FOR EACH ROW

BEGIN

INSERT into viewtable set id=new.employee\_id, f\_name=new.first\_name, l\_name=new.last\_name, salary=new.salary, jo\_date=new.joining\_date, deparment=new.deparment; END $$

1. Create table given below: Salesperson and Customer

## Create table salesperson

CREATE table Salesperson(SNO int PRIMARY KEY

AUTO\_INCREMENT,SNAME CHAR(20),CITY CHAR(30),COMM VARCHAR(20));

## Insert data

INSERT INTO salesperson ( SNAME , CITY , COMM ) VALUES

( peel , london , .12 ), ( serres , sanjose , .13 ), ( motike , london , .11 ),

( rafkin , barcelona , .15 ), ( axeirod , new york , .1 );

# Create table CUSTOMER

CREATE TABLE CUSTOMER(CNM INT PRIMARY KEY,CNAME

CHAR(20),CITY CHAR(30),RATING INT,SNO INT);

## Add foreign

ALTER TABLE customer ADD FOREIGN key(sno) REFERENCES salesperson (sno);

## Insert data

INSERT INTO customer VALUES

(201, Hoffman , London ,100,1001), (202, Giovanne , Roe ,200,1003),

(203, Liu , Aan jose ,300,1002),

(204, Grass , Barcelona ,100,1002),

(206, Clemens , London ,300,1007),

(207, Pereira , Roe ,100,1004);

1. All orders for more than $1000.

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

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

Select SNAME,CITY from Salesperson where COMM>0.12

1. All salespeople either in Barcelona or in Londo

Select \* from Salesperson where CITY= london OR CITY= barcelona ;

1. All salespeople with commission between 0.10 and 0.12.

(Boundary valuesshould be excluded)

Select \* from Salesperson where comm between 0.10 and

0.12;

1. All customers excluding those with rating <= 100 unless they are located inRome

SELECT \* FROM customer WHERE RATING<=100 AND

CITY= roe ;

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

SELECT \* FROM salesperson;

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,order\_date,purch\_amt FROM orders

WHERE salesman\_id=5001;

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 prise 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 and 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