DBMS ASSIGN - 03

```
-- DROP TABLE STUDENTS;
-- DROP TABLE TEACHERS;
-- DROP TABLE GUIDED;
CREATE TABLE STUDENTS(
S_id VARCHAR(20) PRIMARY KEY,
S_name VARCHAR(10) ,
Course VARCHAR(10),
City VARCHAR(20),
State VARCHAR(20),
Mark DECIMAL (7,2)
);
CREATE TABLE TEACHERS(
T_id VARCHAR(20) PRIMARY KEY ,
T_name VARCHAR(10) ,
City VARCHAR(20),
State VARCHAR(20)
);
CREATE TABLE GUIDED(
G_id VARCHAR(20) ,
G_date DATE ,
G_year INT ,
S_id VARCHAR(20),
T_id VARCHAR(20) ,
FOREIGN KEY(S_id) REFERENCES STUDENTS(S_id),
FOREIGN KEY(T_id) REFERENCES TEACHERS(T_id)
);
-- SELECT * FROM STUDENTS
-- ORDER BY S_id DESC;
```

```
INSERT INTO STUDENTS VALUES('21UCS001', 'ROHIT' ,
'B.TECH','KOLKATA','WEST BENGAL',7.8);
INSERT INTO STUDENTS VALUES('21UCS002', 'RAHUL' ,
'B.TECH','GUWAHATI','ASSAM',8.1);
INSERT INTO STUDENTS VALUES('21UCS003', 'VINOD' , 'B.TECH','NEW
DELHI','DELHI',7.5);
INSERT INTO STUDENTS VALUES('21PCS001', 'AJAY' ,
'M.TECH','HYDERABAD','TELANGANA',8.0);
INSERT INTO STUDENTS VALUES('21PCS005', 'MAHESH' ,
'M.TECH','IMPHAL','MANIPUR',7.2);
INSERT INTO STUDENTS VALUES('19DCS001', 'ADITYA' ,
'PH.D','BHUBANESHWAR','ODISHA',9.1);
-- SELECT * FROM TEACHERS
-- ORDER BY State DESC;
INSERT INTO TEACHERS VALUES('20FCS001', 'LALIT' , 'CHENNAI','TAMIL
NADU');
INSERT INTO TEACHERS VALUES('19FCS003', 'AMIT' ,
'HYDERABAD','TELANGANA');
INSERT INTO TEACHERS VALUES('21FCS012', 'VENKAT' ,'GUWAHATI','ASSAM');
-- SELECT * FROM GUIDED;
INSERT INTO GUIDED VALUES('21CS01', '2021-02-
02',2021,'21UCS001','20FCS001');
INSERT INTO GUIDED VALUES('19CS01','2019-06-19' ,
2019,'19DCS001','19FCS003');
INSERT INTO GUIDED VALUES('21CS02', '2021-06-02',
2021,'21UCS002','21FCS012');
INSERT INTO GUIDED VALUES('21CS03', '2021-06-02' ,
2021,'21UCS003','20FCS001');
INSERT INTO GUIDED VALUES('21CS04', '2021-07-02',
2021,'21PCS001','21FCS012');
INSERT INTO GUIDED VALUES('21CS05', '2021-07-02',
```

```
2021,'21PCS005','19FCS003');
-- 1
SELECT G_id,G_year,G_date FROM GUIDED;
-- 2
SELECT * FROM GUIDED WHERE T_id='19FCS003';
-- 3
SELECT city,S_name,State,Mark,S_id,Course FROM STUDENTS;
-- 4
SELECT Mark,S_name FROM students ORDER BY Mark DESC;
-- 5
SELECT * FROM GUIDED WHERE(G_year > 2019 OR NOT (G_date ='2021-06-02'));
-- 6
SELECT S_name , City FROM STUDENTS WHERE State = 'Assam' OR State =
'Delhi' AND Mark>7 ;
-- 7
SELECT * FROM students WHERE NOT Mark <= 8;
-- 9
SELECT S_name , S_id , Mark FROM STUDENTS
WHERE Mark >= 8;
-- 8
CREATE TABLE EMPLOYEE(
ID INT UNSIGNED NOT NULL AUTO_INCREMENT ,
Dept_ID INT ,
Gender VARCHAR(1),
E_Name VARCHAR(20),
E_Salary INT ,
E_ID INT PRIMARY KEY,
KEY(ID)
);
INSERT INTO EMPLOYEE (
Dept_ID ,
```

```
Gender,
E_Name,
E_Salary ,
E_ID
)
VALUES(2,'M','JOHN',1000,123),(4, 'F' ,'BELA', 2000 ,313),( 3, 'F'
,'KATY', 2000, 335),
(1, 'M' ,'RON', 2000, 533),(2, 'M', 'KEN', 1000 ,563),(2, 'M' ,'JOHN',
2000, 321),(4, 'F', 'TAYLOR', 1000, 312),( 3, 'F', 'TAYLOR', 2000, 442),(
3 ,'M', 'RAZOR', 1000, 565);
SELECT DISTINCT E_Name , Gender FROM EMPLOYEE;
```

ASSIGN -04

```
-- DROP TABLE EMPLOYEE;
CREATE TABLE EMPLOYEE(
ID INT UNSIGNED NOT NULL AUTO_INCREMENT ,
Dept_ID INT ,
Gender VARCHAR(1),
E_Name VARCHAR(20),
E_Salary INT ,
E_ID INT PRIMARY KEY,
KEY(ID)
);
INSERT INTO EMPLOYEE(
Dept_ID ,
Gender,
E_Name,
E_Salary ,
```

```
E_ID
)
VALUES(2, 'M' ,'JOHN', 100000 ,124),(4 ,'F', 'BELA', 250000, 313),( 3,
'F' ,'KATY', 250000, 335),
(1, 'M' ,'RON', 205000, 533),(2, 'M', 'KEN', 100000 ,563),(2, 'M'
,'JOHN', 205000, 321),(4, 'F', 'TAYLOR', 100000, 312),( 3, 'F', 'TAYLOR',
300000, 442),( 3 ,'M', 'RAZOR', 100000, 565),( 2, 'F', 'YELEEY' ,150000,
564);
-- i
SELECT E_Name,MAX(E_Salary) AS SECOND_MAX FROM EMPLOYEE WHERE E_SALARY <
(SELECT MAX(E_Salary) FROM EMPLOYEE);
-- ii
SELECT E_Name , (E_Salary/12) AS Monthly_Salaray , E_Salary AS
Annual_Salary FROM EMPLOYEE;
-- iii
SELECT * FROM EMPLOYEE WHERE E_ID = (SELECT MIN(E_ID) FROM EMPLOYEE);
-- iv
SELECT * FROM (SELECT * FROM EMPLOYEE ORDER BY ID DESC LIMIT 5)i
ORDER BY ID ASC;
-- v
SELECT * FROM EMPLOYEE WHERE ID > (SELECT COUNT(ID)/2 FROM EMPLOYEE)
ORDER BY ID;
```

ASSIGN 5

```
-- ASSIGNMENT 5
-- DROP TABLE STUDENTS;
-- DROP TABLE COURSES;
```

```
-- Q1
CREATE TABLE STUDENTS(
S_ID INT UNIQUE AUTO_INCREMENT PRIMARY KEY,
S_Name VARCHAR(20),
S_Address VARCHAR(20),
S_Age INT
);
CREATE TABLE COURSES(
C__ID INT ,
S__ID INT
-- FOREIGN KEY(S__ID) REFERENCES STUDENTS(S_ID)
);
INSERT INTO STUDENTS (
S_Name ,
S_Address,
S_Age
)
VALUES('LOKI', 'DELHI', 19 ),('KISHAN', 'KERELA', 20 ),('RISHI', 'ASSAM',
18),('SANJOY' ,'KOLKATA',18 ),('VISHAL', 'TELANGANA', 20 ),
('PRIYA', 'ASSAM', 19 ),('PURU', 'BIHAR', 18),('RIYA', 'KARNATAKA', 19);
INSERT INTO COURSES (C_ID,S_ID)
VALUES(1,1),(2,2),(2,3),(3,4),(1,5),(4,9),(5,10),(4,11);
-- i
SELECT * FROM STUDENTS INNER JOIN COURSES ON STUDENTS.S_ID =
COURSES.S__ID ORDER BY S_ID ASC;
-- ii
SELECT * FROM STUDENTS LEFT JOIN COURSES ON STUDENTS.S_ID = COURSES.S__ID
ORDER BY S_ID ASC;
-- iii
SELECT * FROM STUDENTS RIGHT JOIN COURSES ON STUDENTS.S_ID =
COURSES.S_ID ORDER BY S_ID ASC;
```

```
-- iv
SELECT * FROM STUDENTS LEFT JOIN COURSES ON STUDENTS.S_ID = COURSES.S__ID
UNION
SELECT * FROM STUDENTS RIGHT JOIN COURSES ON STUDENTS.S_ID =
COURSES.S__ID;
-- Q2
-- DROP TABLE SALARY;
CREATE TABLE SALARY(
ID INT AUTO_INCREMENT,
Emp_Name VARCHAR(20),
Salary INT ,
KEY(ID)
);
INSERT INTO SALARY(
Emp_Name,
Salary
)
VALUES('LOKI',2000),('KISHAN',1500),('RISHI',2000),('SANJOY',3000),('VISH
AL',4000),('PRIYA',1000),('PURU',9000),('RIYA',2000);
SELECT A.ID, A.Salary , B.Emp_Name FROM SALARY AS A LEFT JOIN SALARY AS B
ON A.Salary < B.Salary ORDER BY A.ID ASC;
-- Q3
-- DROP TABLE EMPLOYEE;
-- DROP TABLE DEPARTMENT;
-- DROP TABLE GENDER;
CREATE TABLE EMPLOYEE(
EmpNo INT,
EmpName VARCHAR(20),
Deptid INT,
Genderid INT
);
```

```
CREATE TABLE DEPARTMENT(
Dept_id INT,
DeptName VARCHAR(5)
);
CREATE TABLE GENDER(
Gender_id INT,
Gender VARCHAR(6)
);
INSERT INTO EMPLOYEE(
EmpNo ,
EmpName ,
Deptid,
Genderid
)
VALUES(1886,'John Snow',101,1),(1889,'Amara Giselle',102,2),(1890,'Richie
Tung',102,1);
INSERT INTO DEPARTMENT(
Dept_id,
DeptName
)
VALUES(101,'HR'),(102,'IT');
INSERT INTO GENDER(
Gender_id,
Gender
)
VALUES(01,'Male'),(02,'Female');
SELECT E.EmpNo,E.EmpName,E.Deptid,E.Genderid,D.DeptName,G.Gender
FROM EMPLOYEE AS E
INNER JOIN DEPARTMENT AS D
ON E.Deptid = D.Dept_id
INNER JOIN GENDER AS G
```

```
ON E.Genderid = G.Gender_id
ORDER BY EmpNo ASC;
```

ASSGN : 6 (I)

```
CREATE TABLE TA(
TA_id INT PRIMARY KEY,
Name VARCHAR(20)
);


CREATE TABLE Question(
Q_id INT,
TA__id INT
-- FOREIGN KEY(TA__id) REFERENCES TA(TA_id)
);

INSERT INTO TA (
TA_id,
Name
)
VALUES(5077, 'Rose' ),(21283, 'Angela' ),(62743, 'Frank' ),(88255, 'Patrick' ),(96196, 'Lisa');

INSERT INTO Question (
Q_id ,
TA__id
)
VALUES(61654, 5077 ),(58302, 21283 ),(40587,88255 ),(29477, 5077),(1220, 21283),(69514,
21283),(48561, 62743),(58077, 62743),(18483, 88255),(76766, 21283),(52382, 5077),(74467,
21283),(33625, 96196),(26053, 88255),(42665, 62743),(12859,62743),(70094,
21283),(34599,88255),(54680, 88255),(61881, 5077);
```

```
SELECT TA_id,Name,Qns_created
FROM(
SELECT TA_id,Name,COUNT(Name) AS Qns_created
FROM Question,TA
WHERE TA_id=TA__id
GROUP BY TA_id
ORDER BY Qns_created DESC
)AS T
WHERE Qns_created IN (SELECT T2.cnt FROM(
SELECT MAX(T1.cnts) AS cnt
FROM (SELECT COUNT(*) as cnts
FROM Question
GROUP BY TA__id) AS T1
UNION
SELECT t.cnts
FROM (SELECT COUNT(*) AS cnts
FROM Question
GROUP BY TA__id) t
GROUP BY t.cnts
HAVING COUNT(t.cnts) = 1
)AS T2
);
```

ASSGN : 6 (II)

```
CREATE TABLE TA(
TA_id INT PRIMARY KEY,
Name VARCHAR(20)
);
```

```
CREATE TABLE Question(
Q_id INT,
TA__id INT
);

INSERT INTO TA (
TA_id,
Name
)
VALUES(12299, 'Rose' ),(34856, 'Angela' ),(79345, 'Frank' ),(80491, 'Patrick' ),(81041, 'Lisa');

INSERT INTO Question (
Q_id ,
TA__id
)
VALUES(63963, 81041 ),(63117, 79345 ),(28225,34856 ),(21989, 12299),(4653, 12299),(70070,
79345),(36905, 34856),(61136, 80491),(17234, 12299),(80308, 79345),(40510, 34856),(79820,
80491),(22720, 12299),(22720, 12299),(21394, 12299),(36261,34856),(15334,
12299),(23157,34856),(54102, 34856),(69065, 80491);

SELECT TA_id,Name,Qns_created
FROM(
SELECT TA_id,Name,COUNT(Name) AS Qns_created
FROM Question,TA
WHERE TA_id=TA__id
GROUP BY TA_id
ORDER BY Qns_created DESC
)AS T
WHERE Qns_created IN (SELECT T2.cnt FROM(
SELECT MAX(T1.cnts) AS cnt
FROM (SELECT COUNT(*) as cnts
FROM Question
```

```
GROUP BY TA__id) AS T1
UNION
SELECT t.cnts
FROM (SELECT COUNT(*) AS cnts
FROM Question
GROUP BY TA__id) t
GROUP BY t.cnts
HAVING COUNT(t.cnts) = 1
)AS T2
);
```

ASSGN : 7 (I)

```
CREATE TABLE Rooms(
R_id INT PRIMARY KEY,
RoomName VARCHAR(20)
);

CREATE TABLE  Employees(
E_id INT PRIMARY KEY,
LastName VARCHAR(20),
FirstName VARCHAR(20)
);

CREATE TABLE Assignments(
A_id INT,
Room INT,
E_id INT,
Days INT,
```

```
FOREIGN KEY(Room) REFERENCES Rooms(R_id),
FOREIGN KEY(E_id) REFERENCES Employees(E_id)
);


INSERT INTO Rooms (
R_id ,
RoomName
)
VALUES(1, 'Pantry' ),(2, 'Conference');

INSERT INTO Employees (
E_id ,
LastName ,
FirstName
)
VALUES(1,'Zufall' ,'Rainer' ),(2, 'Piper','Peter');

INSERT INTO Assignments(
A_id ,
Room ,
E_id ,
Days
)
VALUES(1,2,1,2),(2,2,1,3),(3,2,2,4),(4,2,1,5),(5,2,1,6),(6,1,2,2),(7,1,2,4),(8,1,2,6);



SELECT  T1.Days,Conference,Pantry FROM
(
SELECT FirstName AS Conference,Days FROM Employees E,Assignments A,Rooms R
```

```
WHERE Room=2 AND E.E_id=A.E_id
GROUP BY Days
)AS T1
LEFT JOIN
(
SELECT FirstName AS Pantry,Days FROM Employees E,Assignments A,Rooms R
WHERE Room=1 AND E.E_id=A.E_id
GROUP BY Days
)AS T2 ON T1.Days = T2.Days
ORDER BY T1.Days ASC;
```

ASSGN : 7 (II)

```
CREATE TABLE Customer(
customer_id INT,
cust_name VARCHAR(20),
city VARCHAR(20),
grade INT,
salesman_id INT
);

CREATE TABLE  Salesman(
salesman_id INT PRIMARY KEY,
name VARCHAR(20),
city VARCHAR(20),
commission DECIMAL(2,2)
);

CREATE TABLE Orderss(
```

```
ord_no INT PRIMARY KEY,
purch_amt DECIMAL(10,2),
ord_date DATE,
customer_id INT,
salesman_id INT,
FOREIGN KEY (salesman_id) REFERENCES Salesman(salesman_id)
);



INSERT INTO Customer(
customer_id ,
cust_name,
city ,
grade ,
salesman_id
)
VALUES(3002,'Nick Rimando','New York',100,5001),(3007,'Brad Davis','New
York',200,5001),(3005,'Graham Zusi','California',200,5002),(3008,'Julian
Green','London',300,5002),(3004,'Fabian Johnson','Paris',300,5006),(3009,'Geoff
Cameron','Paris',300,5006),(3009,'Geoff
Cameron','Berlin',100,5003),(3003,'Jozy Altidor','Moscow',200,5007),(3001,'Brad
Guzan','London',100,5005);



INSERT INTO Salesman(
salesman_id ,
name ,
city ,
commission
)
```

```
VALUES(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);



INSERT INTO Orderss(
ord_no ,
purch_amt ,
ord_date,
customer_id ,
salesman_id
)
VALUES(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',3004,5003),
(70003,2480.4,'2012-10-10',3009,5002),
(70012,250.45,'2012-06-27',3008,5006),
(70011,75.29,'2012-08-17',3003,5007),
(70013,3045.6,'2012-04-25',3002,5001);

SELECT a.cust_name,a.city,a.grade,
b.name AS "Salesman",
c.ord_no, c.ord_date, c.purch_amt
FROM customer a
RIGHT OUTER JOIN salesman b
ON b.salesman_id=a.salesman_id
```

```
LEFT OUTER JOIN orderss c
ON c.customer_id=a.customer_id
WHERE c.purch_amt>=2000
AND a.grade IS NOT NULL;
```

ASSIGN 8

```
-- i
SELECT * FROM Customer
WHERE Country = 'Germany' AND city = 'Berlin' OR city = 'Munchen';

-- ii
SELECT * FROM Customer
WHERE NOT Country = 'Germany' AND NOT Country='USA';

-- iii
DELETE FROM Customer
WHERE Customer_name = 'Alfreds Futterkiste';

-- iv
UPDATE Customer
SET Contact_name = 'Prabhaaa', city = 'visakhapatnam' WHERE Customer_id=1;

-- v
SELECT TOP 3 * FROM Customer;

-- vi
SELECT COUNT(price),SUM(price),AVG(price) FROM Products;
```

```
-- vii
  WHERE Customer_name LIKE 'a%';
  WHERE Customer_name LIKE '%a';
  WHERE Customer_name LIKE '%or%';
  WHERE Customer_name LIKE '_r%';
  WHERE Customer_name LIKE 'a_%';
  WHERE Customer_name LIKE 'a%e';

-- viii
  SELECT OrderID,Quantity,
  CASE
        WHEN Quantity > 20 THEN 'The quantity is greater than 20'
    WHEN Quantity < 20 THEN 'The quantity is less than 20'
    ELSE 'The quantity is 20'
  END AS QuantityT
  FROM OrderDetails;

-- ix
  CREATE TABLE Person(
    ID INT PRIMARY KEY,
    NAME VARCHAR(20),
    ADDRESS VARCHAR(20)
    );

    INSERT INTO Person(
     ID,NAME,ADDRESS
     )VALUES(1,'Raju','Mumbai');

 -- x
```

```
CREATE TABLE First(
 S_id INT PRIMARY KEY,
 LastName VARCHAR(10),
 FirstName VARCHAR(10),
 City VARCHAR(10)
 );


 CREATE TABLE Second(
 O_id INT PRIMARY KEY,
 OrderNo INT,
 S_id INT,
 CONSTRAINT fk_id
        FOREIGN KEY (S_id)
        REFERENCES First (S_id)
        ON DELETE SET NULL
 );


 INSERT INTO First(
 S_id ,
 LastName,
 FirstName,
 City

)VALUES(1,'MAURYA','AJEET','ALLAHABAD'),(2,'JAISWAL','RATAN','GHAZIABAD'),(3,'ARORA','SAUMYA
','MODINAGAR');


 INSERT INTO Second(
 O_id ,
 OrderNo ,
```

```
S_id
)VALUES(1,99586465,2),(2,78466588,2),(3,22354846,3),(4,57698656,1);

SELECT * FROM First;
SELECT * FROM Second;

DELETE FROM First WHERE S_id=2;

SELECT * FROM First;
SELECT * FROM Second;
```