Ron Andrew J – 23 MAY MYSQL TEST

create database test;

use test;

create table worker(worker\_id int primary key, first\_name varchar(15),

last\_name varchar(15), salary int, joining\_date datetime,

department varchar(15));

insert into worker values

(001,'Monika','Arora',100000,'2014-02-20 09:00:00','HR'),

(002, 'Niharika','Verma',80000,'2014-06-11 09:00:00', 'Admin'),

(003,'Vishal','Singhal',300000,'2014-02-20 09:00:00','HR'),

(004, 'Amitabh','Singh',500000,'2014-02-20 09:00:00', 'Admin'),

(005, 'Vivek','Bhati',500000,'2014-06-11 09:00:00', 'Admin'),

(006, 'Vipul','Diwan',200000,'2014-06-11 09:00:00', 'Account'),

(007, 'Satish','Kumar',75000,'2014-01-20 09:00:00', 'Account'),

(008, 'Geetika','Chauhan',90000,'2014-04-11 09:00:00', 'Admin');

CREATE TABLE BONUS(worker\_ref\_id int, bonus\_date datetime, bonus\_amount int,

CONSTRAINT BONUS\_FK FOREIGN KEY (worker\_ref\_id) REFERENCES worker(worker\_id));

INSERT INTO BONUS VALUES

(1,'2016-02-20 00:00:00',5000),

(2,'2016-06-11 00:00:00',3000),

(3,'2016-02-20 00:00:00',4000),

(1,'2016-02-20 00:00:00',4500),

(2,'2016-06-11 00:00:00',3500);

create table TITLE(worker\_ref\_id int, worker\_title varchar(15), affected\_from datetime,

constraint TITLE\_FK foreign key (worker\_ref\_id) references WORKER(worker\_id));

INSERT INTO TITLE VALUES

(1,'Manager','2016-02-20 00:00:00'),

(2,'Executive','2016-06-11 00:00:00'),

(8,'Executive','2016-06-11 00:00:00'),

(5,'Manager','2016-06-11 00:00:00'),

(4,'Asst. Manager','2016-06-11 00:00:00'),

(7,'Executive','2016-06-11 00:00:00'),

(6,'Lead','2016-06-11 00:00:00'),

(3,'Lead','2016-06-11 00:00:00');

**#1. Write an SQL query to fetch unique values of DEPARTMENT from Worker table.**

SELECT DISTINCT(DEPARTMENT) FROM WORKER;

**#2. Write an SQL query to print all Worker details from the Worker table order by FIRST\_NAME Ascending and DEPARTMENT Descending**

SELECT \* FROM WORKER ORDER BY FIRST\_NAME ASC, DEPARTMENT DESC;

**#3. Write an SQL query to print details of the Workers whose FIRST\_NAME contains ‘a’**

SELECT \* FROM WORKER WHERE FIRST\_NAME LIKE '%a%';

**#4. Write an SQL query to print details of the Workers whose FIRST\_NAME ends with ‘h’ and contains six alphabets**

SELECT \* FROM WORKER WHERE FIRST\_NAME LIKE '\_\_\_\_\_h';

**#5. Write an SQL query to print details of the Workers whose SALARY lies between 100000 and 500000**

SELECT \* FROM WORKER WHERE SALARY BETWEEN 100000 AND 500000;

**#6. Write an SQL query to print details of the Workers who have joined in Feb’2014.**

SELECT \* FROM WORKER WHERE JOINING\_DATE LIKE '2014-02%';

**#7. Write an SQL query to fetch the count of employees working in the department ‘Admin’**

SELECT DEPARTMENT, COUNT(\*) FROM WORKER WHERE DEPARTMENT='Admin' GROUP BY DEPARTMENT;

**#8. Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.**

SELECT FIRST\_NAME, LAST\_NAME FROM WORKER WHERE SALARY BETWEEN 50000 AND 100000;

**#9. Write an SQL query to fetch the no. of workers for each department in the descending order**

SELECT DEPARTMENT, COUNT(\*) FROM WORKER GROUP BY DEPARTMENT ORDER BY COUNT(\*) DESC;

**#10. Write an SQL query to print details of the Workers who are also Managers**

SELECT \* FROM WORKER INNER JOIN TITLE ON worker\_id=worker\_ref\_id where worker\_title='Manager';

**#11. Write an SQL query to determine the 2nd lowest salary without using TOP or limit method.**

SELECT SALARY FROM WORKER W1 WHERE 1=(SELECT COUNT(DISTINCT(SALARY)) FROM WORKER W2 WHERE W1.SALARY>W2.SALARY);

**#12. Write an SQL query to fetch the list of employees with the same salary**

SELECT FIRST\_NAME, SALARY FROM WORKER W1 WHERE

(SELECT COUNT(DISTINCT(SALARY)) FROM WORKER W2 WHERE W1.SALARY=W2.SALARY AND NOT W1.worker\_id=W2.worker\_id)>0;

**#13. Write an SQL query to show the second highest salary from a table**

SELECT DISTINCT(SALARY) FROM WORKER ORDER BY SALARY DESC LIMIT 1 OFFSET 1;

**#14. Write an SQL query to show one row twice in results from a table.**

SELECT \* FROM WORKER

UNION ALL

SELECT \* FROM WORKER;

**#15. Write an SQL query to fetch the first 50% records from a table.**

SELECT \* FROM WORKER LIMIT 4;

**#16. Write an SQL query to fetch the departments that have less than three people in it.**

SELECT DEPARTMENT, COUNT(\*) FROM WORKER GROUP BY DEPARTMENT HAVING COUNT(\*)<3;

**#17. Write an SQL query to show all departments along with the number of people in there.**

SELECT DEPARTMENT, COUNT(\*) FROM WORKER GROUP BY DEPARTMENT;

**#18. Write an SQL query to fetch the last five records from a table**

SELECT \* FROM WORKER LIMIT 5 OFFSET 3;

**#19. Write an SQL query to print the name of employees having the highest salary in each department**

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT, SALARY FROM WORKER W1 WHERE

SALARY=(SELECT MAX(SALARY) FROM WORKER W2 WHERE W1.DEPARTMENT=W2.DEPARTMENT);

**#20. Write an SQL query to fetch three max salaries from a table**

SELECT DISTINCT(SALARY) FROM WORKER ORDER BY SALARY DESC LIMIT 3;

**#21. Write an SQL query to print the name of employees having the lowest salary in accunt and admin department**

SELECT FIRST\_NAME, LAST\_NAME, Department, salary FROM WORKER W1

WHERE SALARY=(SELECT MIN(SALARY) FROM WORKER W2 WHERE W1.DEPARTMENT=W2.DEPARTMENT) AND DEPARTMENT IN ('Account','Admin');