MYSQL TEST

# CREATE DATABASE ORG123; SHOW DATABASES;

USE ORG123;

CREATE TABLE Worker (

# WORKER\_ID INT NOT NULL PRIMARY KEY AUTO\_INCREMENT, FIRST\_NAME CHAR(25),

LAST\_NAME CHAR(25), SALARY INT(15), JOINING\_DATE DATETIME, DEPARTMENT CHAR(25)

);

INSERT INTO Worker

# (WORKER\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) VALUES

(001, 'Monika', 'Arora', 100000, '14-02-20 09.00.00', 'HR'),

(002, 'Niharika', 'Verma', 80000, '14-06-11 09.00.00', 'Admin'),

(003, 'Vishal', 'Singhal', 300000, '14-02-20 09.00.00', 'HR'),

(004, 'Amitabh', 'Singh', 500000, '14-02-20 09.00.00', 'Admin'),

(005, 'Vivek', 'Bhati', 500000, '14-06-11 09.00.00', 'Admin'),

(006, 'Vipul', 'Diwan', 200000, '14-06-11 09.00.00', 'Account'),

(007, 'Satish', 'Kumar', 75000, '14-01-20 09.00.00', 'Account'),

(008, 'Geetika', 'Chauhan', 90000, '14-04-11 09.00.00', 'Admin');

CREATE TABLE Bonus ( WORKER\_REF\_ID INT, BONUS\_AMOUNT INT(10), BONUS\_DATE DATETIME, FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID) ON DELETE CASCADE

);

INSERT INTO Bonus

# (WORKER\_REF\_ID, BONUS\_AMOUNT, BONUS\_DATE) VALUES (001, 5000, '16-02-20'),

(002, 3000, '16-06-11'),

(003, 4000, '16-02-20'),

(001, 4500, '16-02-20'),

(002, 3500, '16-06-11'); CREATE TABLE Title (

# WORKER\_REF\_ID INT, WORKER\_TITLE CHAR(25), AFFECTED\_FROM DATETIME, FOREIGN KEY (WORKER\_REF\_ID)

REFERENCES Worker(WORKER\_ID) ON DELETE CASCADE

);

INSERT INTO Title

(WORKER\_REF\_ID, WORKER\_TITLE, AFFECTED\_FROM) VALUES (001, 'Manager', '2016-02-20 00:00:00'),

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

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

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

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

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

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

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

use ORG123;

#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 MONTH(JOINING\_DATE) = 2 AND YEAR(JOINING\_DATE) = 2014;

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

SELECT COUNT(\*) AS Admin\_Worker\_Count FROM Worker WHERE DEPARTMENT

= 'Admin';

#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(\*) AS Worker\_Count FROM Worker GROUP BY DEPARTMENT ORDER BY Worker\_Count DESC;

#10. Write an SQL query to print details of the Workers who are also Managers SELECT \* FROM Worker WHERE WORKER\_ID IN ( SELECT WORKER\_REF\_ID FROM Title WHERE WORKER\_TITLE = 'Manager');

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

SELECT MIN(SALARY) AS Second\_Lowest\_Salary FROM Worker WHERE SALARY > ( SELECT MIN(SALARY) FROM Worker);

#12. Write an SQL query to fetch the list of employees with the same salary SELECT \* FROM Worker WHERE SALARY IN ( SELECT SALARY FROM Worker GROUP BY SALARY HAVING COUNT(\*) > 1);

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

SELECT MAX(SALARY) AS Second\_Highest\_Salary FROM Worker WHERE SALARY

< ( SELECT MAX(SALARY) FROM Worker);

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

SELECT \* FROM Worker WHERE WORKER\_ID = 1 UNION ALL SELECT \* FROM Worker WHERE WORKER\_ID = 1;

#15.Write an SQL query to fetch the first 50% records from a table. SELECT \* FROM Worker ORDER BY WORKER\_ID LIMIT 4;

#16. Write an SQL query to fetch the departments that have less than three people in it. SELECT DEPARTMENT 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(\*) AS COUNT\_DEPT FROM Worker GROUP BY DEPARTMENT;

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

SELECT \* FROM ( SELECT \* FROM Worker ORDER BY WORKER\_ID DESC LIMIT 5)

AS last\_five

# ORDER BY WORKER\_ID ASC;

#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 w WHERE SALARY = (

SELECT MAX(SALARY) FROM Worker WHERE DEPARTMENT = w.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 w WHERE DEPARTMENT IN ('Account', 'Admin') AND SALARY = (

SELECT MIN(SALARY) FROM Worker WHERE DEPARTMENT = w.DEPARTMENT);