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');

select \* from Worker;

select \* from Bonus;

select \* from Title;

#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(\*) from worker where deparment = '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 total\_workers

from worker

group by department

order by total\_workers desc;

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

select worker.\*

from worker

inner join title

on worker.worker\_id = title.worker\_ref\_id

where title.worker\_title = 'manager';

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

select minimum(salary) as second\_lowest\_salary

from worker

where salary > (

select minimum(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 count(\*) from worker;

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 number\_of\_people

from worker

group by department;

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

select \*

from worker

order by worker\_id asc

limit 5;

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

select worker.first\_name

from worker

inner join (

select department, max(salary) as max\_salary

from worker

group by department

) as dept\_max

on worker.department = dept\_max.department and worker.salary = dept\_max.max\_salary;

#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 worker.first\_name

from worker

inner join (

select department, min(salary) as max\_salary

from worker

where department="account" or department="admin"

group by department

) as dept\_max

on worker.department = dept\_max.department and worker.salary = dept\_max.max\_salary;