DROP DATABASE IF EXISTS ORG123;

CREATE DATABASE ORG123;

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, '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\_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, '2016-02-20'),

(002, 3000, '2016-06-11'),

(003, 4000, '2016-02-20'),

(001, 4500, '2016-02-20'),

(002, 3500, '2016-06-11');

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');

# 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' AND LENGTH(TRIM(FIRST\_NAME)) = 6;

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

# 7. write an sql query to fetch the count of employees working in the department

SELECT DEPARTMENT, COUNT(\*) AS EmployeeCount FROM Worker 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(\*) AS WorkerCount FROM Worker GROUP BY DEPARTMENT ORDER BY WorkerCount DESC;

# 10. write an sql query to print details of the workers who are also managers

SELECT W.\* FROM Worker W JOIN Title T ON W.WORKER\_ID = T.WORKER\_REF\_ID WHERE T.WORKER\_TITLE = 'Manager';

# 11. write an sql query to determine the 2nd lowest salary without using top or limit method

SELECT MIN(SALARY) AS SecondLowestSalary 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) ORDER BY SALARY;

# 13. write an sql query to show the second highest salary from a table

SELECT MAX(SALARY) AS SecondHighestSalary 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 (SELECT FLOOR(COUNT(\*) / 2) FROM Worker);

# 16. write an sql query to fetch the departments that have less than three people in it

SELECT DEPARTMENT, COUNT(\*) AS EmployeeCount 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 EmployeeCount FROM Worker GROUP BY DEPARTMENT;

# 18. write an sql query to fetch the last five records from a table in ascending order

SELECT \* FROM (SELECT \* FROM Worker ORDER BY WORKER\_ID DESC LIMIT 5) AS LastFive 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 account 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);