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');

#QUES 1

SELECT DEPARTMENT FROM Worker

GROUP BY DEPARTMENT;

# The SELECT DISTINCT clause ensures that only unique (non-duplicate) values from the DEPARTMENT column are retrieved.

#QUES 2

SELECT \* FROM Worker

ORDER BY FIRST\_NAME ASC, DEPARTMENT DESC;

# SELECT \* retrieves all columns for every worker. The ORDER BY sorts results first by FIRST\_NAME(ascending), and for workers with the same FIRST\_NAME, it sorts them by DEPARTMENT in (descending).

#QUES 3

SELECT \* FROM Worker

WHERE FIRST\_NAME LIKE '%a%';

# '%a%' to select only those FIRST\_NAMEs that have the letter 'a' anywhere within their string.

#QUES 4

SELECT \* FROM Worker

WHERE FIRST\_NAME LIKE '\_\_\_\_\_\_' AND FIRST\_NAME LIKE '%h';

# 6 ubderscores ensures the 6 characters in name and %h ensures name ends with h

#QUES 5

SELECT \* FROM Worker

WHERE SALARY >= 100000 AND SALARY <= 500000;

# AND operator is used to test both the conditions

#QUES 6

SELECT \* FROM Worker

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

# YEAR and MONTH clause to check the joining date

#QUES 7

SELECT COUNT(\*) FROM Worker

WHERE DEPARTMENT = 'Admin';

# WHERE clause checks for department

#QUES 8

SELECT FIRST\_NAME, LAST\_NAME

FROM Worker

WHERE SALARY >= 50000 AND SALARY <= 100000;

# AND clause is used to find the results that lies in between

#QUES 9

SELECT DEPARTMENT, COUNT(WORKER\_ID) AS NumberOfWorkers

FROM Worker

GROUP BY DEPARTMENT

ORDER BY NumberOfWorkers DESC;

# groups the departments and count the workers in each group

#QUES 10

SELECT \* FROM Worker WHERE WORKER\_ID IN

(SELECT WORKER\_REF\_ID FROM Title

WHERE WORKER\_TITLE = 'Manager');

# combines rows of worker and title where there ids match.

#QUES 11

SELECT MIN(SALARY)

FROM Worker

WHERE SALARY > (SELECT MIN(SALARY) FROM Worker);

# inner sub query finds the lowest salary in worker table.

#QUES 12

SELECT \* FROM Worker

WHERE SALARY IN (

SELECT SALARY

FROM Worker

GROUP BY SALARY

HAVING COUNT(SALARY) > 1);

# inner clause finds all salary values that are more than onece in worker table

#QUES 13

SELECT MAX(SALARY)

FROM Worker

WHERE SALARY < (SELECT MAX(SALARY) FROM Worker);

# the inner query finds the highest salary

#QUES 14

SELECT \* FROM Worker

WHERE WORKER\_ID = 1

UNION ALL

SELECT \* FROM Worker

WHERE WORKER\_ID = 1;

# 1st SELECT gets all the columns of worker id=1. 2nd select also does the same. Union combines all the results.

#QUES 15

# We will calculate the total number of rows,

# divide it by two , and then use that number with LIMIT.

#QUES 16

SELECT DEPARTMENT, COUNT(WORKER\_ID) AS NumberOfWorkers

FROM Worker

GROUP BY DEPARTMENT

HAVING COUNT(WORKER\_ID) < 3; #includes departments where the calculated count of workers is less than 3

#QUES 17

SELECT DEPARTMENT, COUNT(WORKER\_ID) AS NumberOfWorkers # selects name of each department and count the number of worker ids

FROM Worker

GROUP BY DEPARTMENT;

#QUES 18

SELECT \* FROM Worker

ORDER BY WORKER\_ID ASC

LIMIT 5; # gets only 1st 5 rows from the sorted results

#QUES 19

#QUES 20

#QUES 21

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT, SALARY

FROM Worker

WHERE DEPARTMENT IN ('Account', 'Admin')

AND SALARY = (

SELECT MIN(SALARY) #selects the lowest salary where department is account and admin

FROM Worker

WHERE DEPARTMENT IN ('Account', 'Admin')

);