

The worker is not having any relation It’s an independent table.

Bonus and Title tables are dependent on the worker table.

A black and white table with numbers and numbers

Description automatically generated

A black and white screen with white text

Description automatically generated

1. Created worker table

create table worker (worker\_id int auto\_increment primary key , first\_name varchar(225),last\_name varchar(225),salary int , joining\_date datetime,department varchar(225));

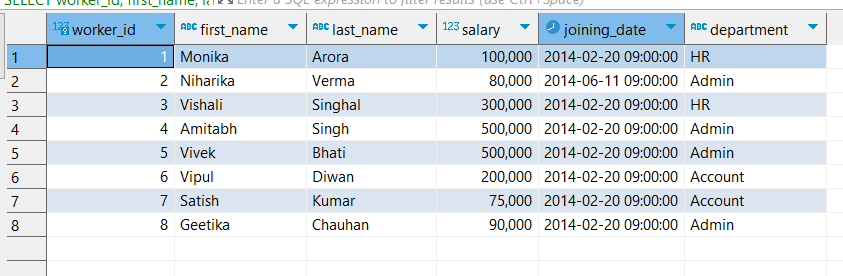
A black screen with white text

Description automatically generated

1. Insert some values in this table.

**insert** **into** org.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');**

****

**\*\* knowledge purpose update query**

**update** org.worker **set** department =**"Account"** **where** worker\_id =006 **or** worker\_id =007;

1. Now let’s create title table

**create** **table** title (worker\_ref\_id **int** ,worker\_title **varchar**(225), affected\_from **datetime** , **foreign** **key**(worker\_ref\_id) **references** worker(worker\_id));

**insert** **into** org.title (worker\_ref\_id ,worker\_title , affected\_from )

**values** (1,**"Manager"**,**"16-02-20 00.00.00"**);

**insert** **into** org.title (worker\_ref\_id ,worker\_title , affected\_from )

**values** (2,**"Executive"**,**"16-06-11 00.00.00"**),

(8,**"Executive"**,**"16-06-11 00.00.00"**),

(5,**"Manager"**,**"16-06-11 00.00.00"**),

(4,**"Asst. Manager"**,**"16-06-11 00.00.00"**),

(7,**"Executive"**,**"16-06-11 00.00.00"**);

A screenshot of a computer

Description automatically generated

Similarly create table of bonus and insert some data in that too.

**create** **table** bonus (worker\_ref\_id **int** ,bonus\_amount **int** , bonus\_date **datetime**,**foreign** **key** (worker\_ref\_id) **references** worker(worker\_id));

**insert** **into** bonus (worker\_ref\_id,bonus\_amount,bonus\_date) **values**

(1,5000,**"16-02-20 00.00.00"**),

(2,3000,**"16-06-11 00.00.00"**),

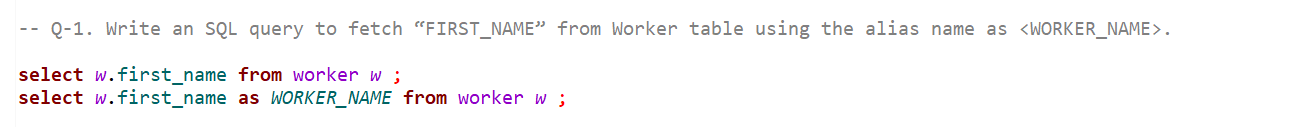
(3,4000,**"16-02-20 00.00.00"**),

(1,4500,**"16-02-20 00.00.00"**),

(2,3500,**"16-06-11 00.00.00"**);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

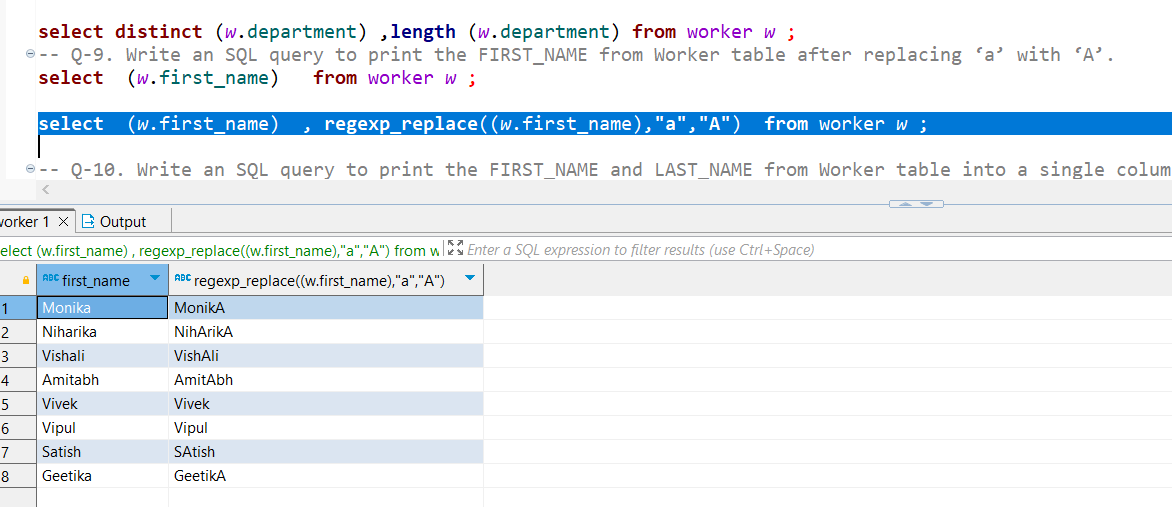
Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated



OR

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

27 mins 🡪 not good .

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A computer screen with numbers and text

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A computer screen with numbers and numbers

Description automatically generatedA computer screen shot of a black screen

Description automatically generated

-- Q-23. Write an SQL query to fetch the no. of workers for each department in the descending order.

A black screen with white text

Description automatically generated

-- Q-23. Write an SQL query to fetch the no. of workers for each department in the descending order.

A screenshot of a computer

Description automatically generated

A computer screen with many lines

Description automatically generated with medium confidence

* Q-25. Write an SQL query to fetch a number (more than 1) of the same titles in the ORG of different types.

A computer screen shot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated



If even records

-- Q-27. Write an SQL query to show only even rows from a table.

A computer screen with numbers and lines

Description automatically generated

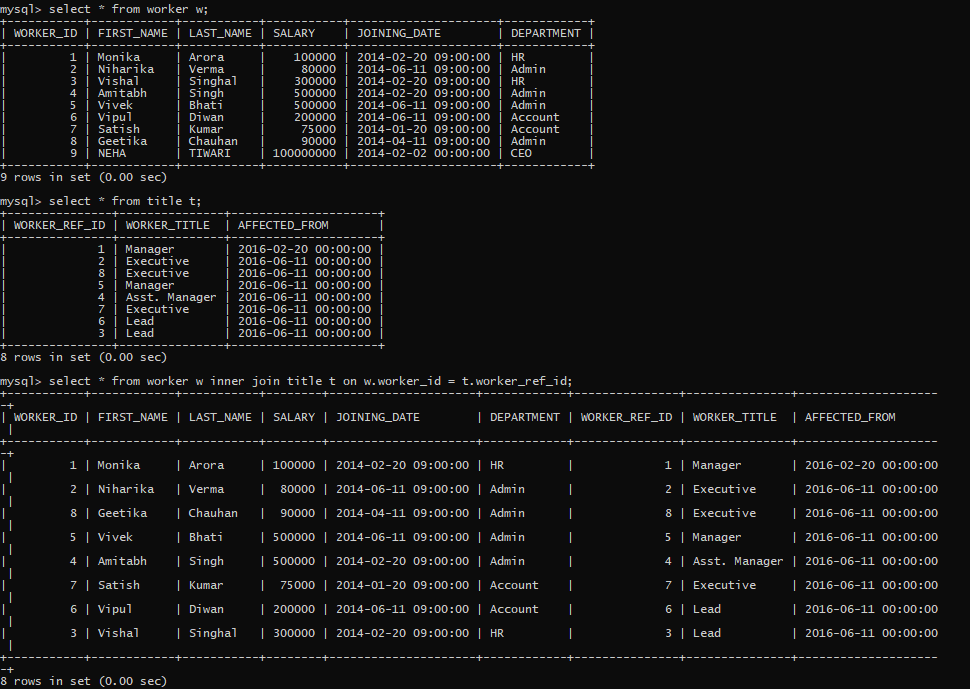
\*\*-- Q-28. Write an SQL query to clone a new table from another table.

1. Cloning first we need to create that table.
2. Then insert all the values from that table.

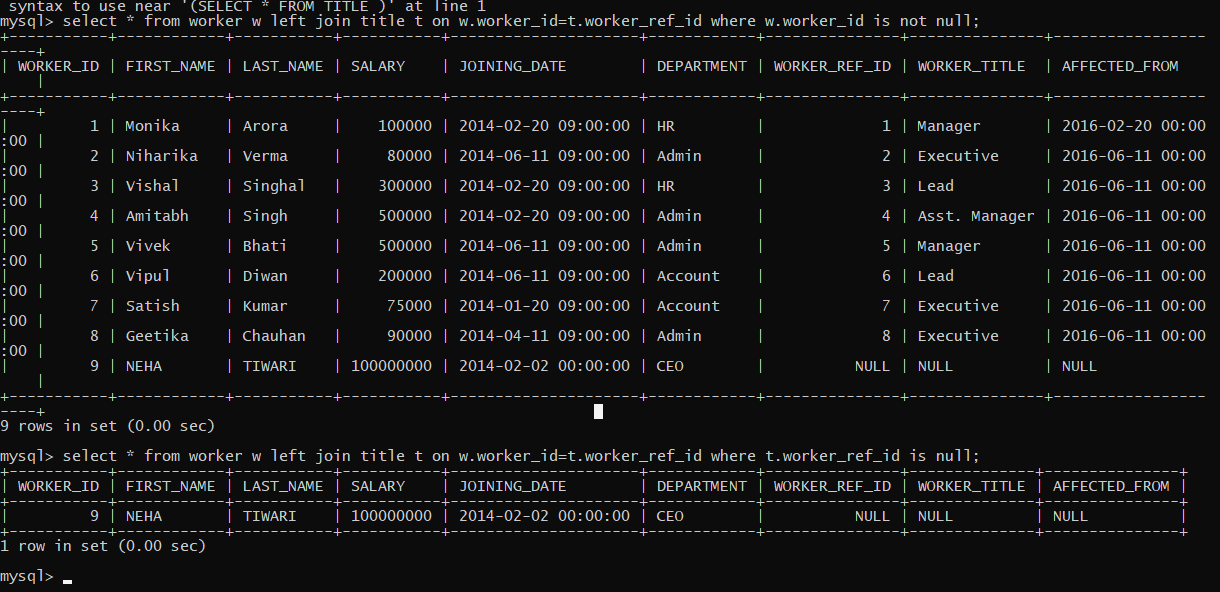
A screenshot of a computer program

Description automatically generated

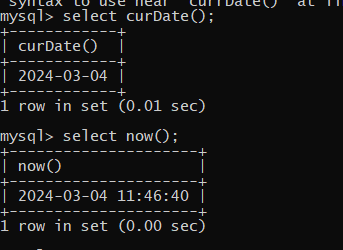
-- Q-29. Write an SQL query to fetch intersecting records of two tables.

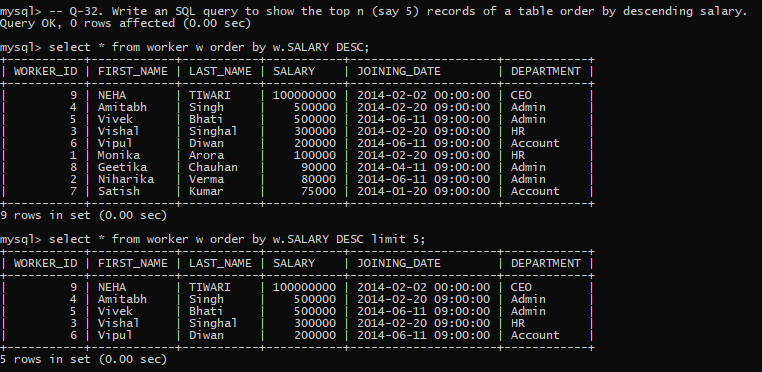


-- Q-30. Write an SQL query to show records from one table that another table does not have.

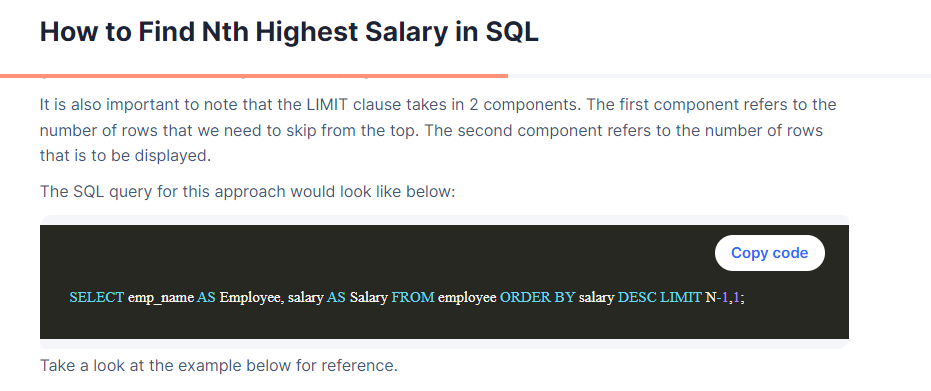


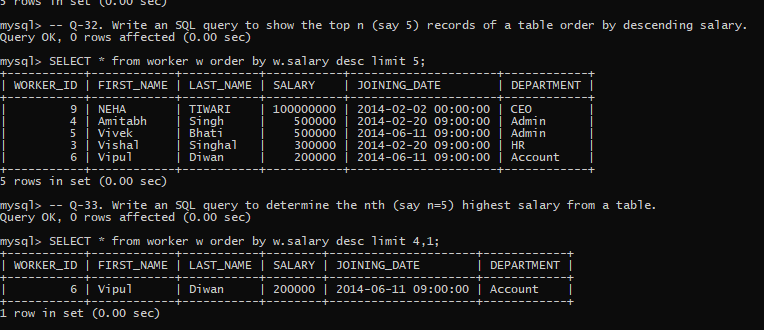
-- Q-31. Write an SQL query to show the current date and time.



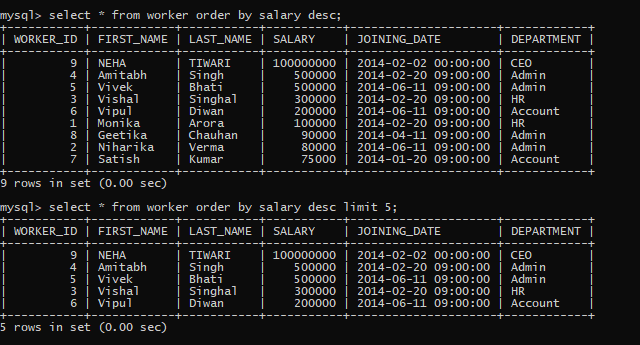


TOP NO 🡺 LIMIT NO.

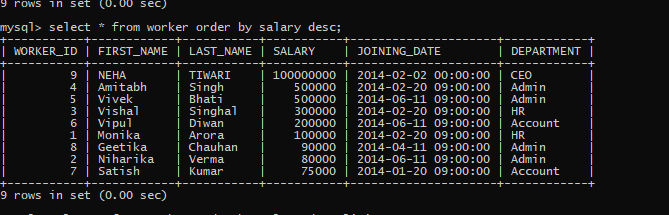


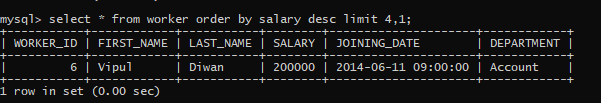


-- Q-32. Write an SQL query to show the top n (say 5) records of a table order by descending salary.

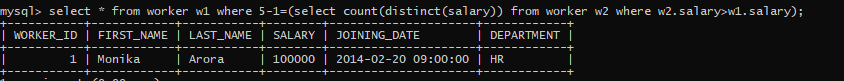


-- Q-33. Write an SQL query to determine the nth (say n=5) highest salary from a table.



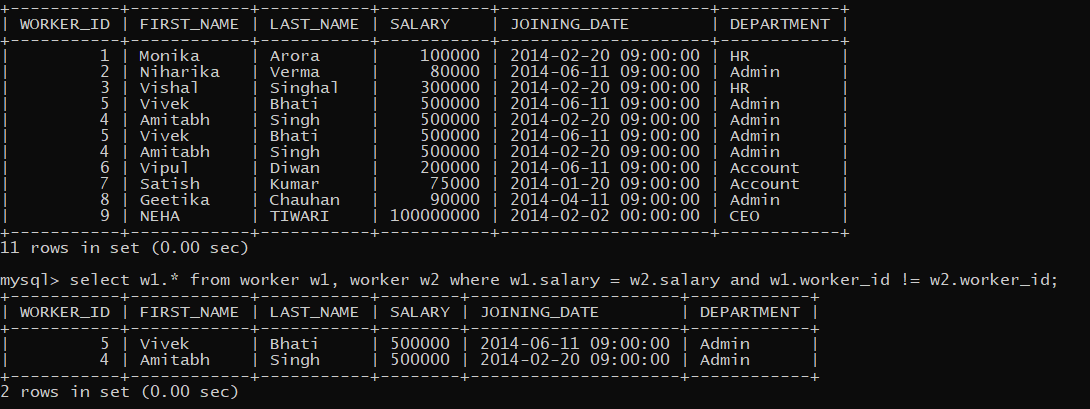
(n-1) this is not distinct .

-- Q-34. Write an SQL query to determine the 5th highest salary without using LIMIT keyword.



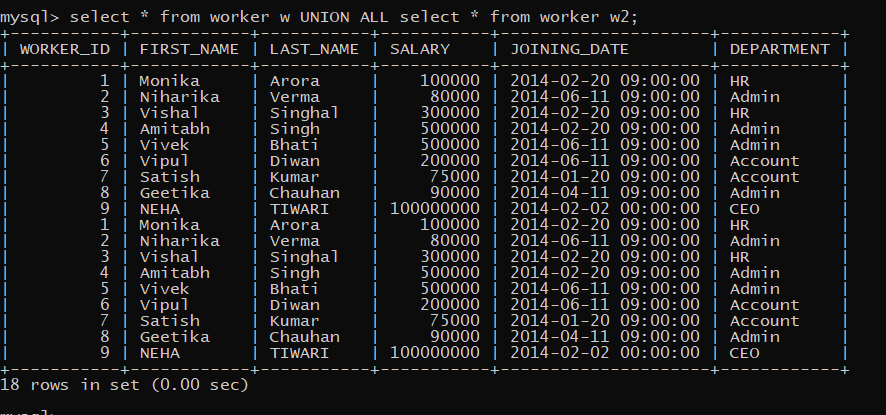
-- Q-35. Write an SQL query to fetch the list of employees with the same salary.

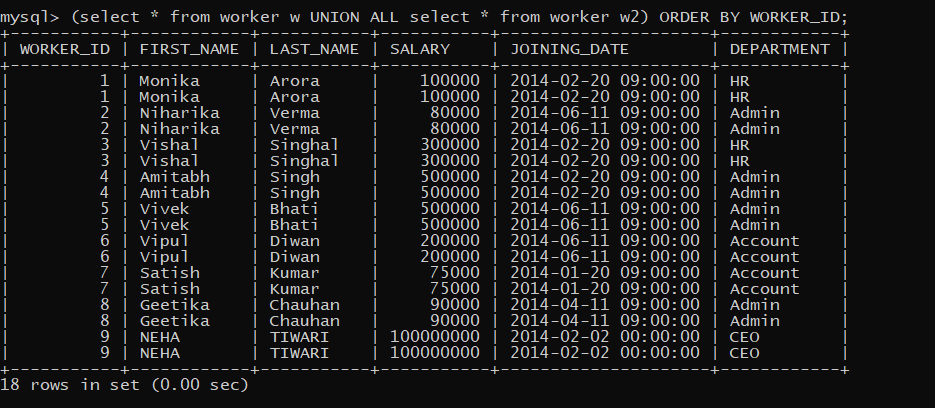
Comparing with self-table.



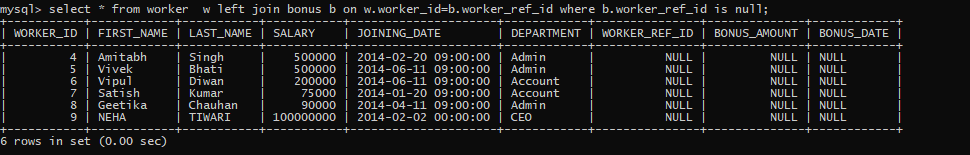
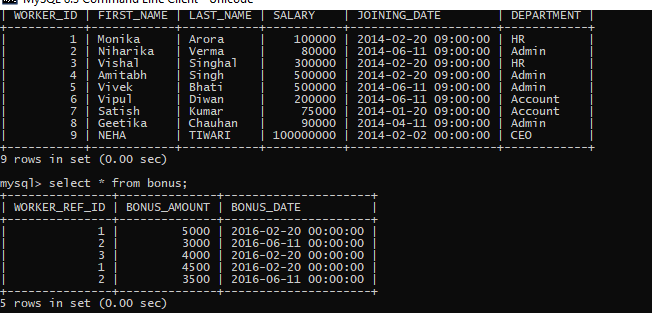
-- Q-36. Write an SQL query to show the second highest salary from a table using sub-query.

-- Q-37. Write an SQL query to show one row twice in results from a table.





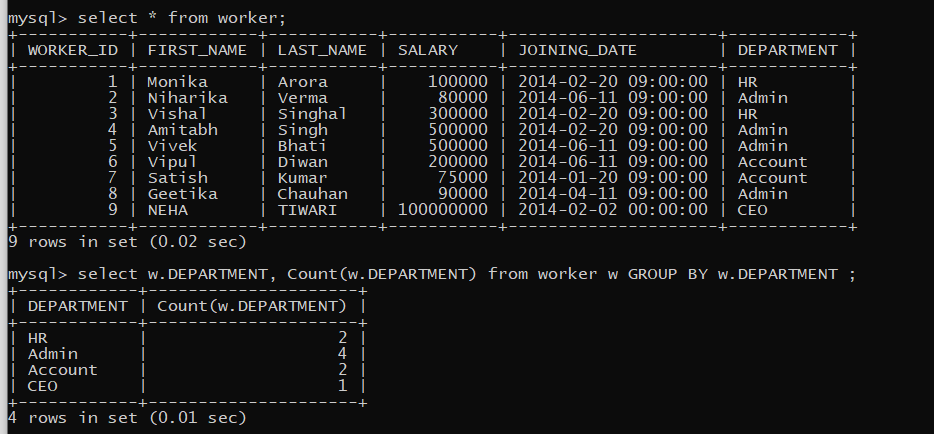
-- Q-38. Write an SQL query to list worker\_id who does not get bonus.



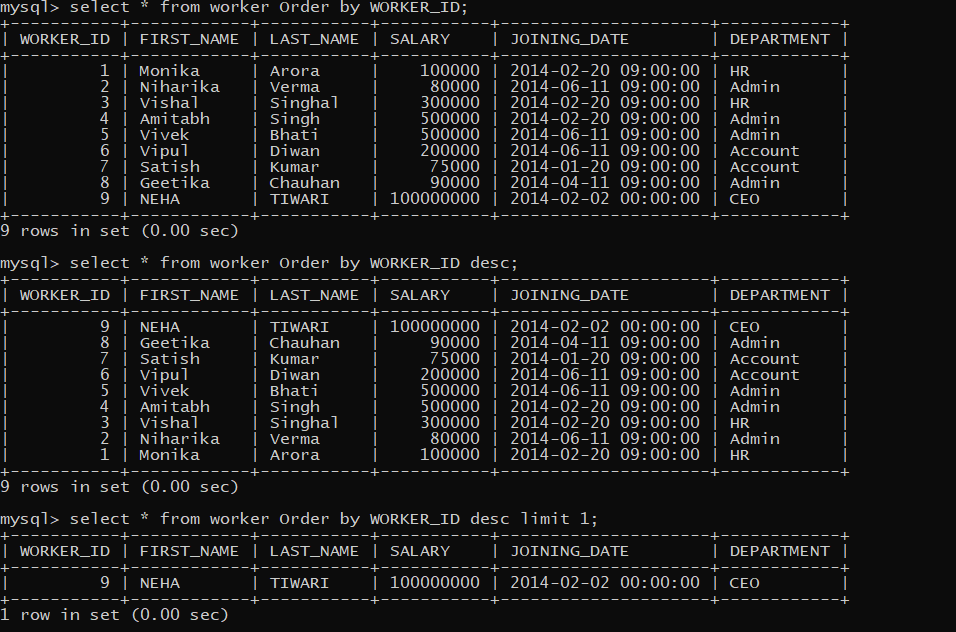
-- Q-39. Write an SQL query to fetch the first 50% records from a table.

-- Q-40. Write an SQL query to fetch the departments that have less than 4 people in it.

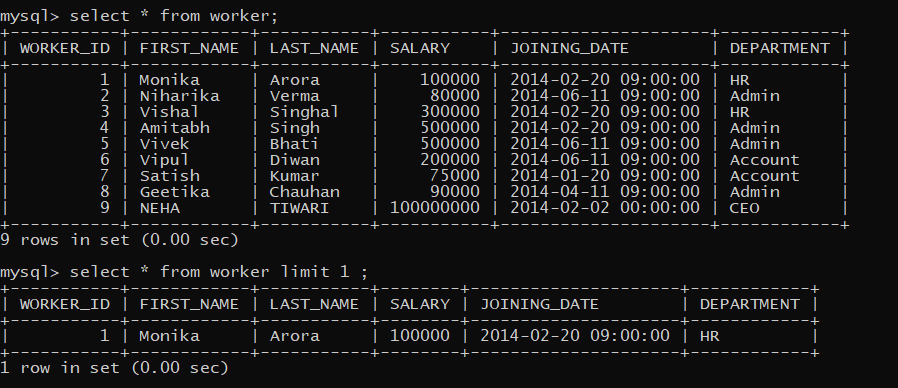
-- Q-41. Write an SQL query to show all departments along with the number of people in there.



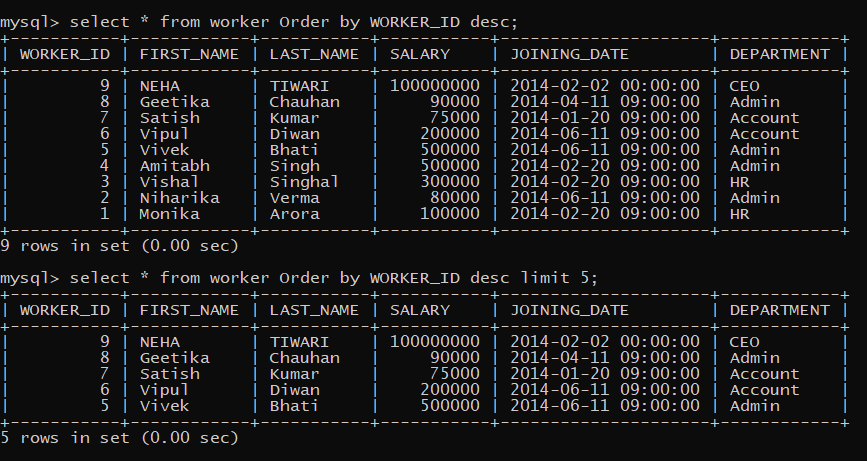
-- Q-42. Write an SQL query to show the last record from a table.



-- Q-43. Write an SQL query to fetch the first row of a table.



-- Q-44. Write an SQL query to fetch the last five records from a table.



-- Q-45. Write an SQL query to print the name of employees having the highest salary in each department.

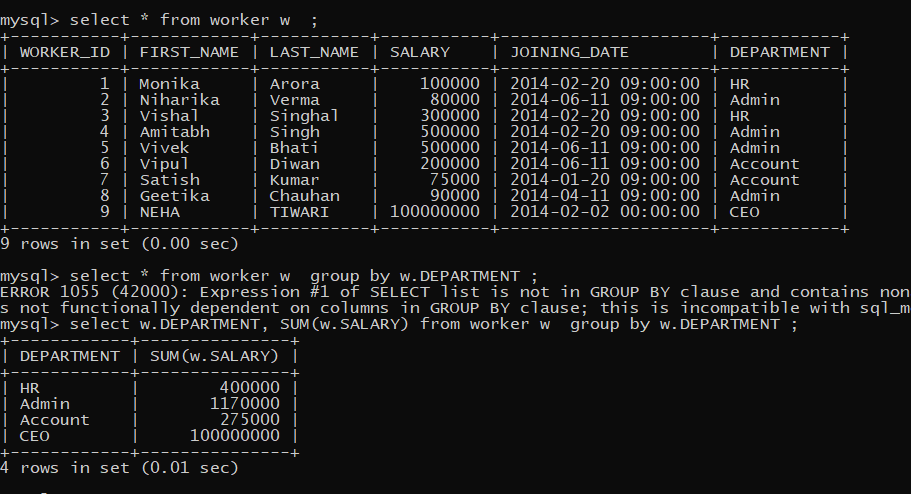
-- Q-46. Write an SQL query to fetch three max salaries from a table using co-related subquery

-- DRY RUN AFTER REVISING THE CORELATED SUBQUERY CONCEPT FROM LEC-9.

-- Q-47. Write an SQL query to fetch three min salaries from a table using co-related subquery

-- Q-48. Write an SQL query to fetch nth max salaries from a table.

-- Q-49. Write an SQL query to fetch departments along with the total salaries paid for each of them.



-- Q-50. Write an SQL query to fetch the names of workers who earn the highest salary.

