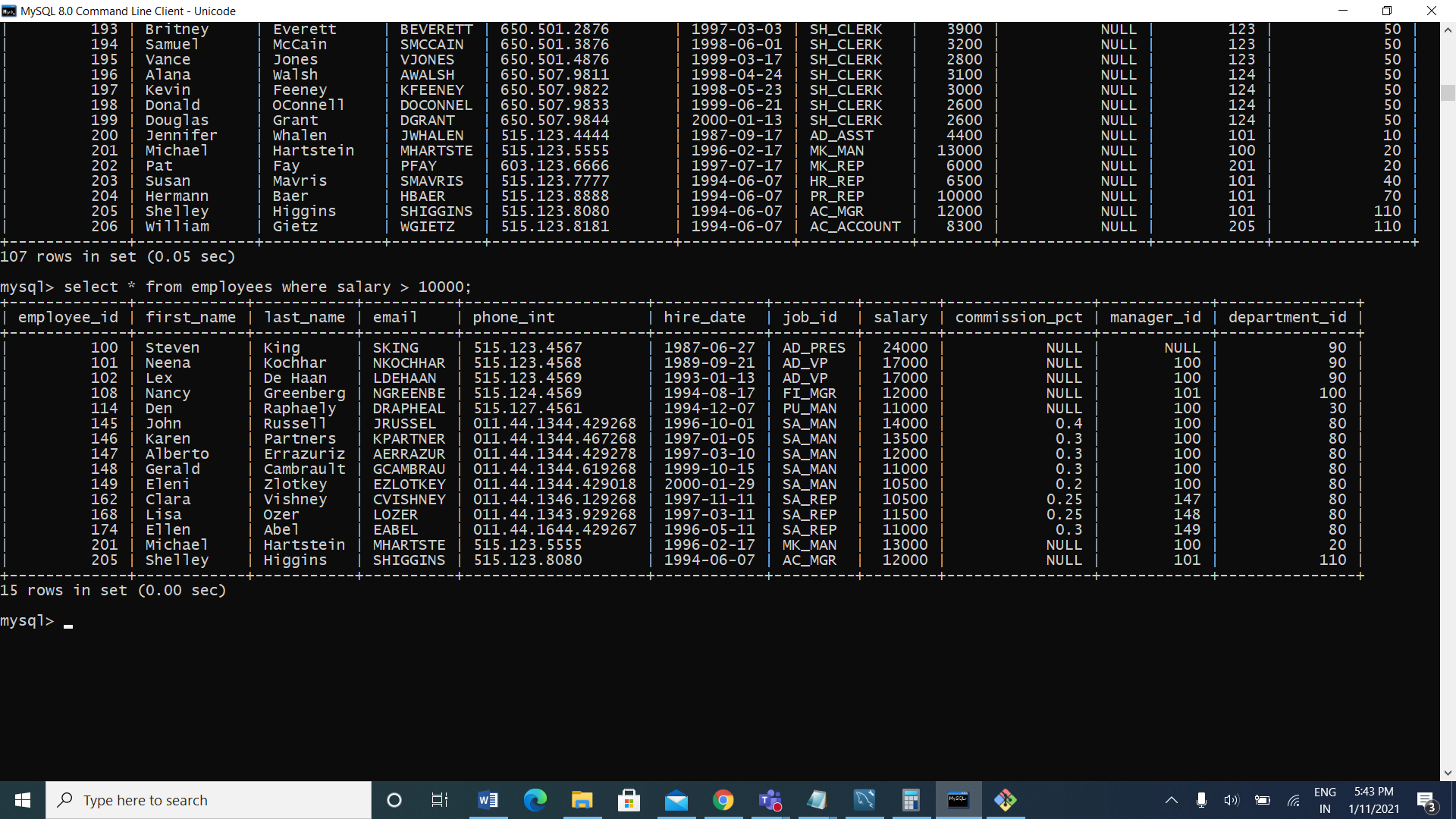
Day 1 (11-01-2021)

Select Clause with Where clause:

1. Display details of jobs where the minimum salary is greater than 10000.

Ans: select \* from employees where salary > 10000;

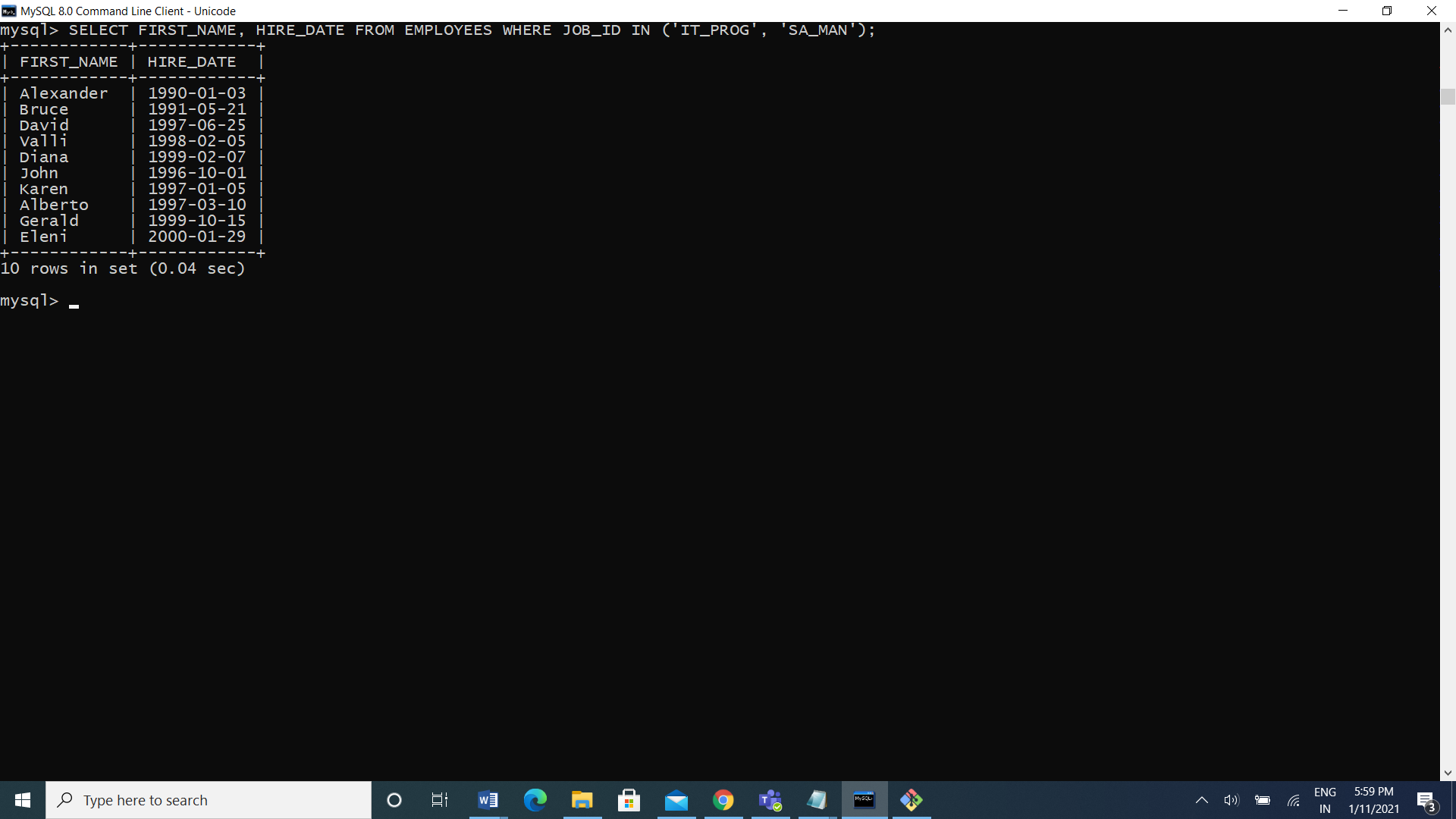


1. Display the first name and join date of the employees who joined between 2002 and 2005.

Ans: select job\_title,department\_name,last\_name,start\_date from job\_history join jobs using (job\_id)join departments using (department\_id) join employees using (employee\_id) where start\_date between 2000 and 2005;

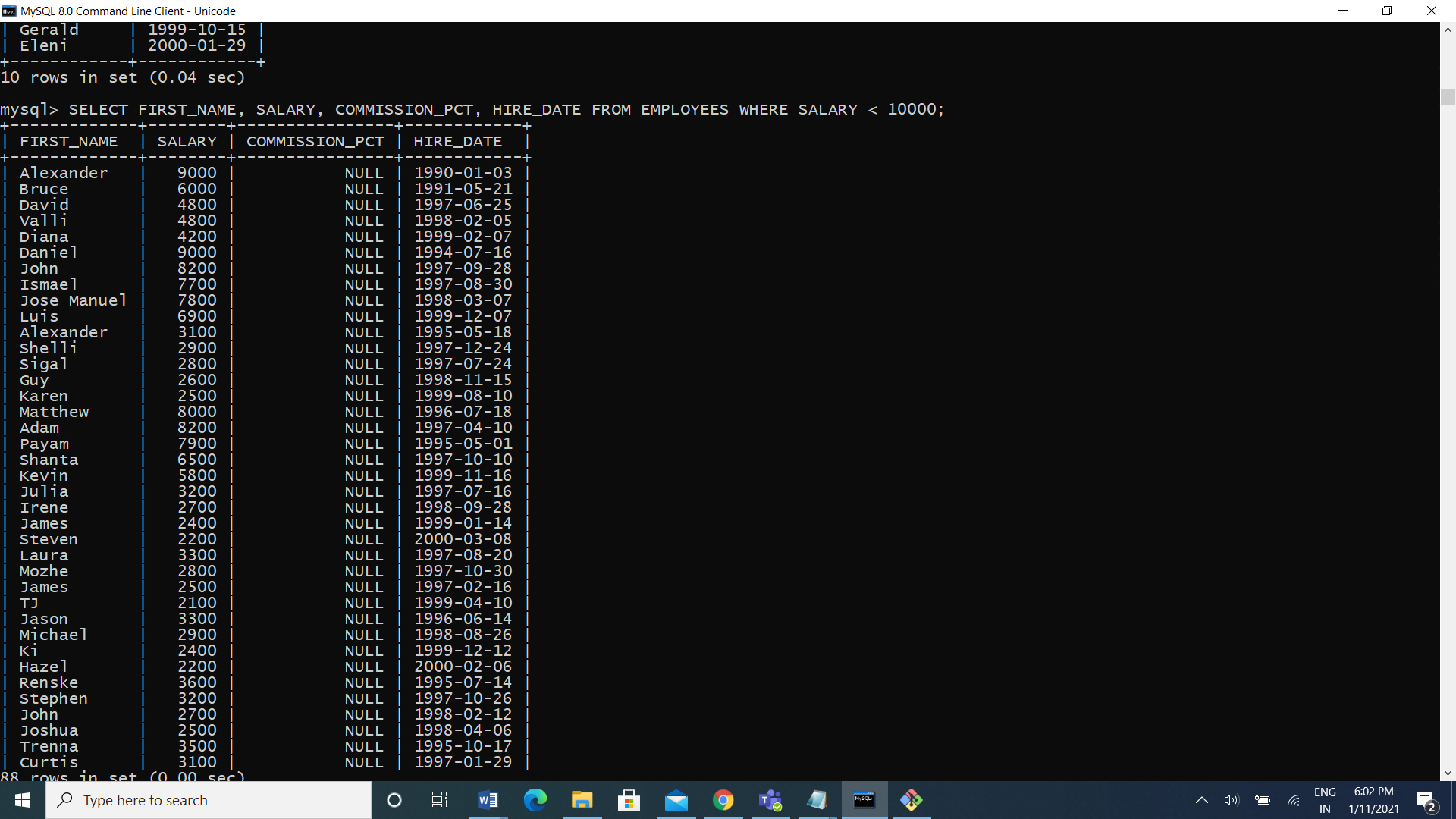
1. Display first name and join date of the employees who is either IT Programmer or Sales Man

Ans: SELECT FIRST\_NAME, HIRE\_DATE FROM EMPLOYEES WHERE JOB\_ID IN ('IT\_PROG', 'SA\_MAN');



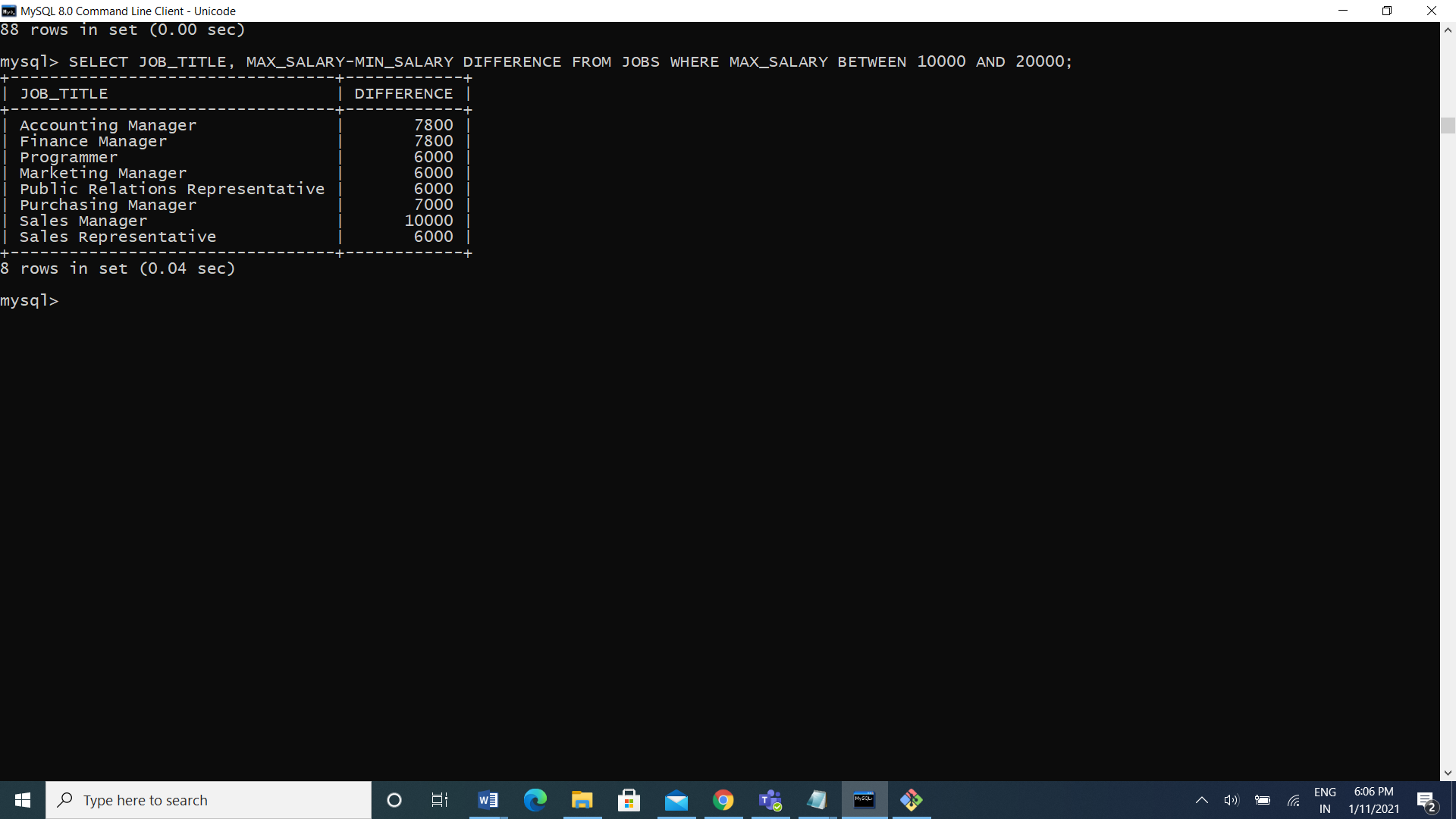
4. Display first name, salary, commission pct, and hire date for employees with salary less than 10000.

Ans: select first\_name, salary, commission\_pct,hire\_date from employees where salary <10000.

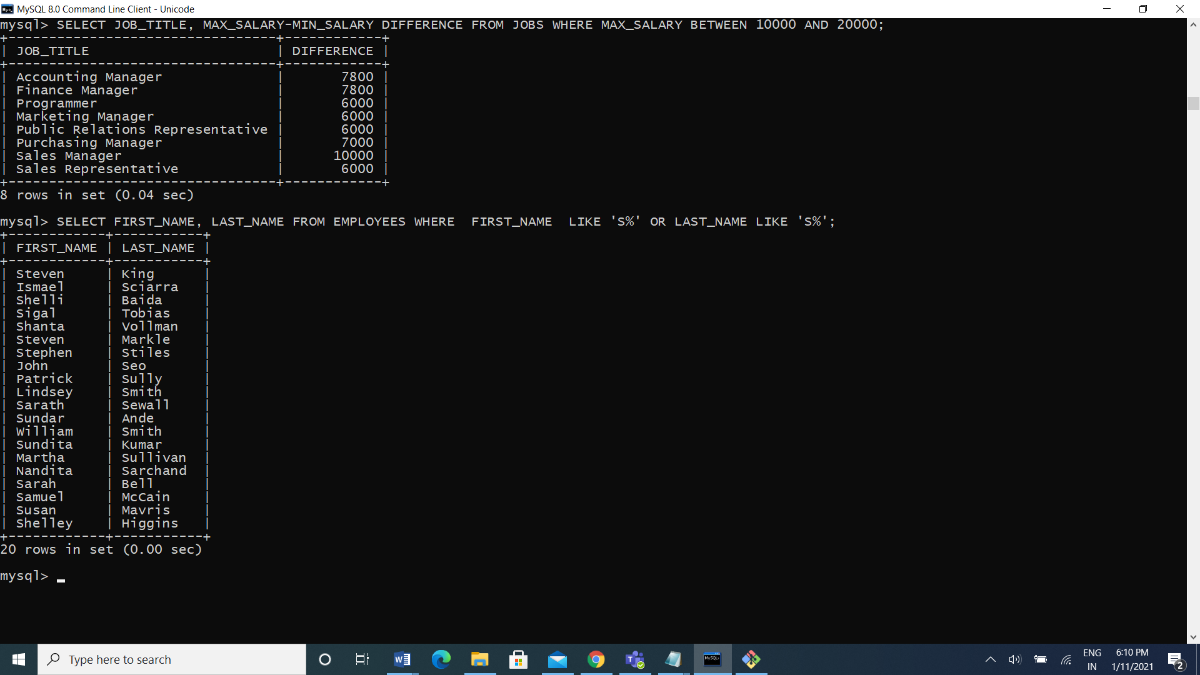


5. Display job Title, the difference between minimum and maximum salaries for jobs with max salary in the range 10000 to 20000.

Ans: select job\_title, max\_salary-min\_salary difference from jobs where max\_salary between 10000 and 20000;

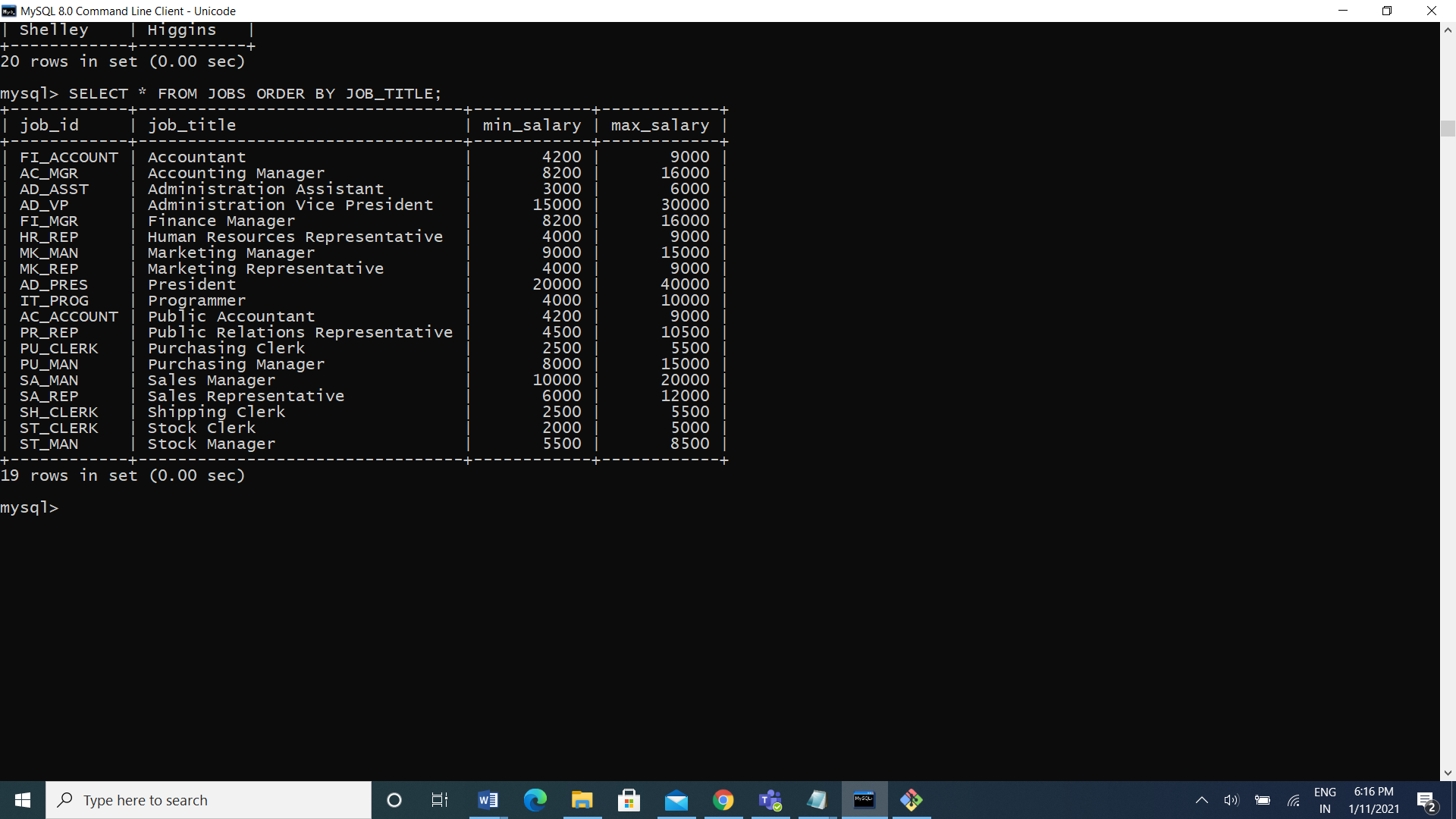


6) Display employees where the first name or last name starts with S.

Ans: select first\_name, last\_name from employees where first\_name like ‘s%’ or last\_name like ‘s%’ 

7) Display details of jobs in the descending order of the title.

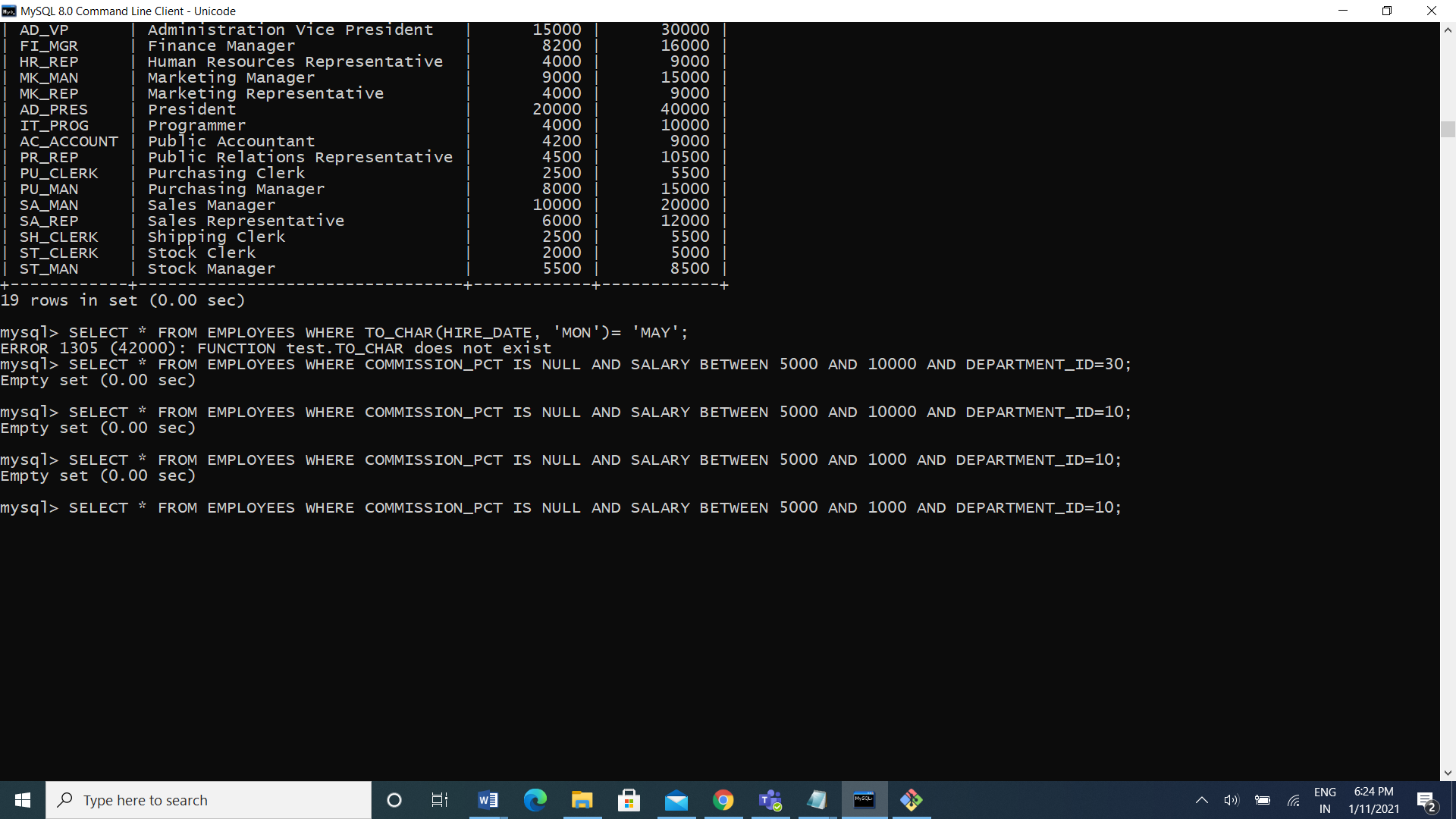
Ans: select \* from job order by job\_title;



8) Display employees who joined in the month of May.

Ans: select

9) Display details of the employees where commission percentage is null and salary in the range 5000 to 10000 and department is 30.

Ans: SELECT \* FROM EMPLOYEES WHERE COMMISSION\_PCT IS NULL AND SALARY BETWEEN 5000 AND 10000 AND DEPARTMENT\_ID=30; 

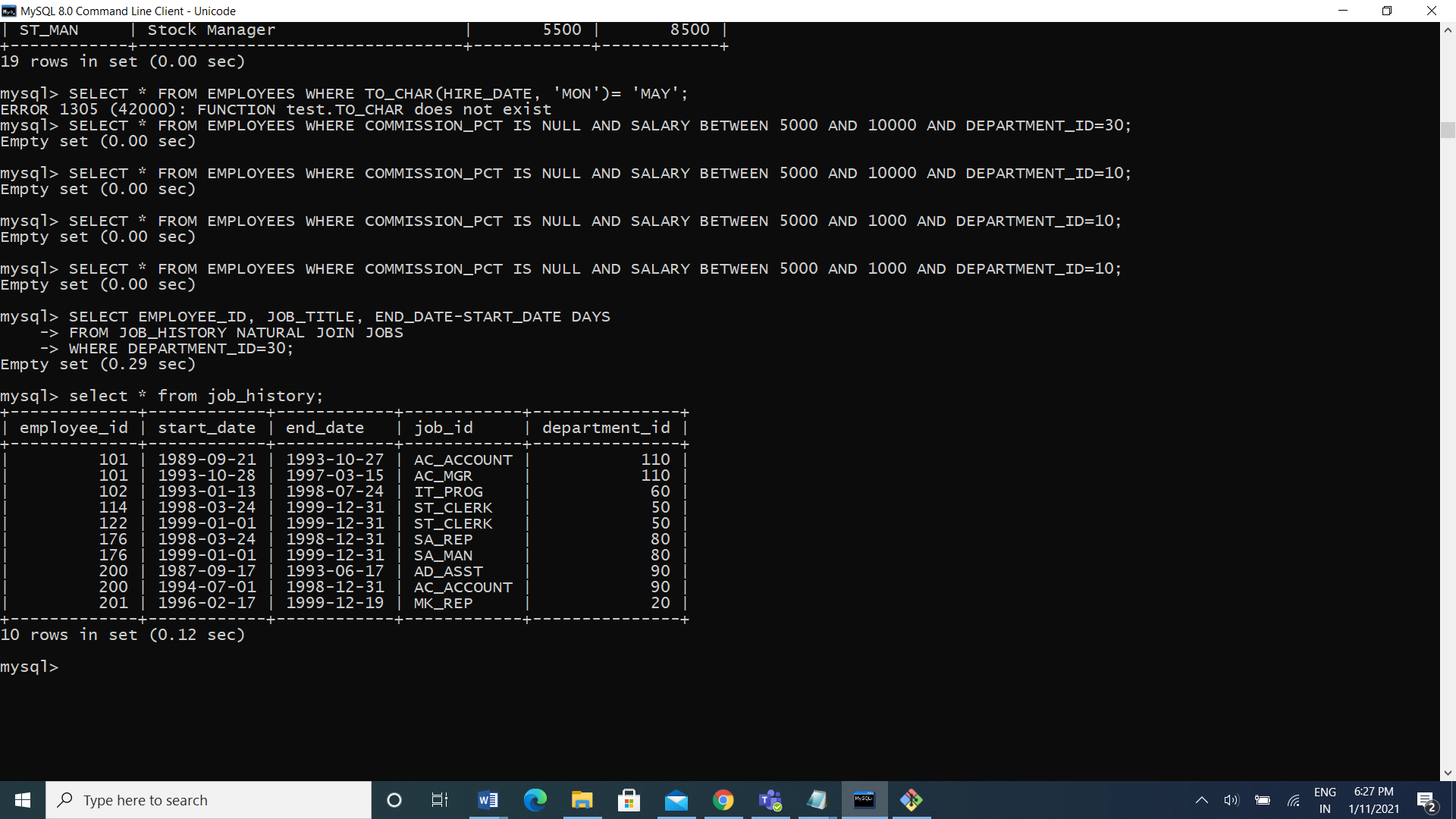
Joints

1). Display job title, employee ID, number of days between ending date and starting date for all jobs in department 30 from job history.

Ans: SELECT EMPLOYEE\_ID, JOB\_TITLE, END\_DATE-START\_DATE DAYS

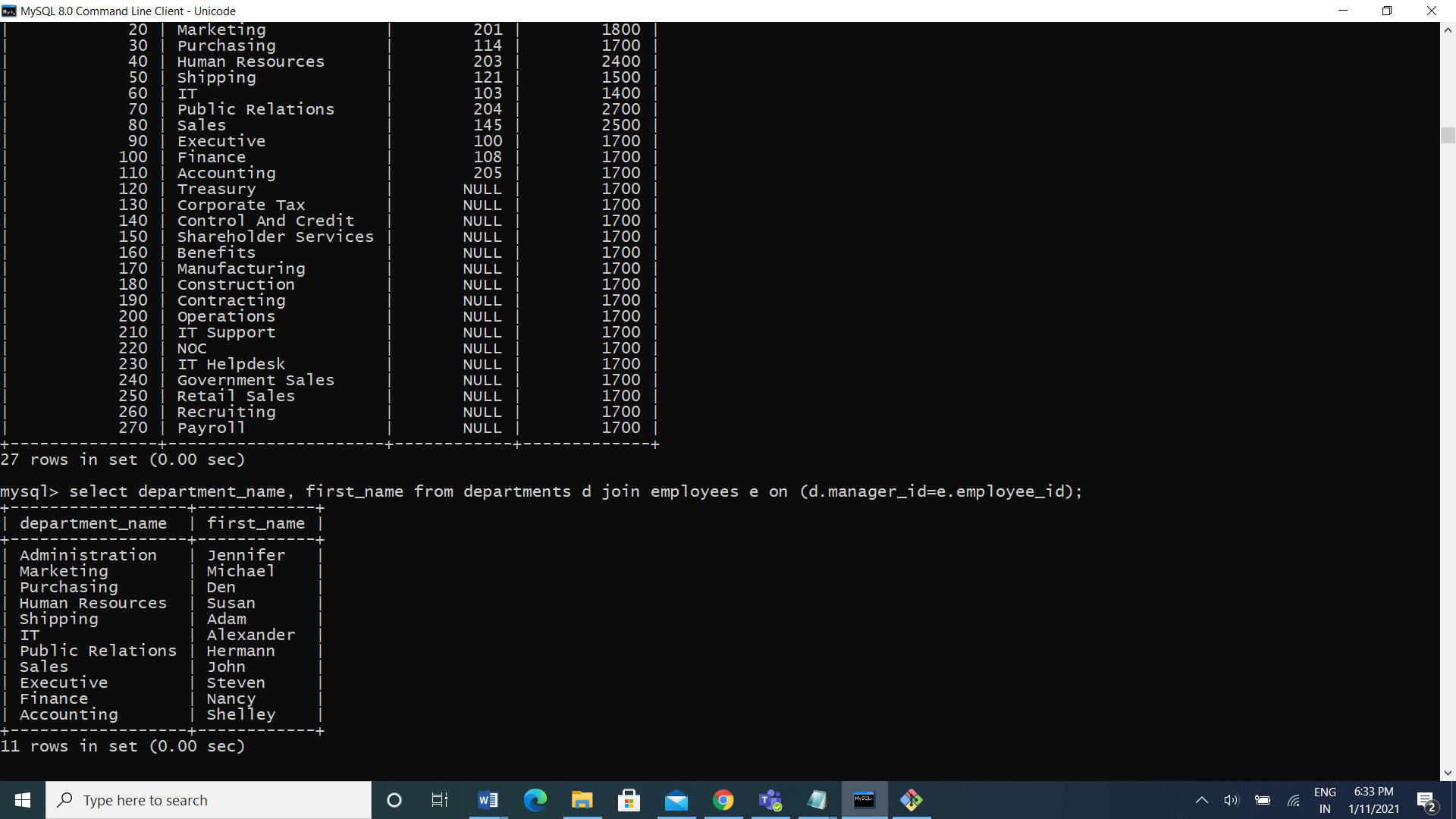
FROM JOB\_HISTORY NATURAL JOIN JOBS

WHERE DEPARTMENT\_ID=30;



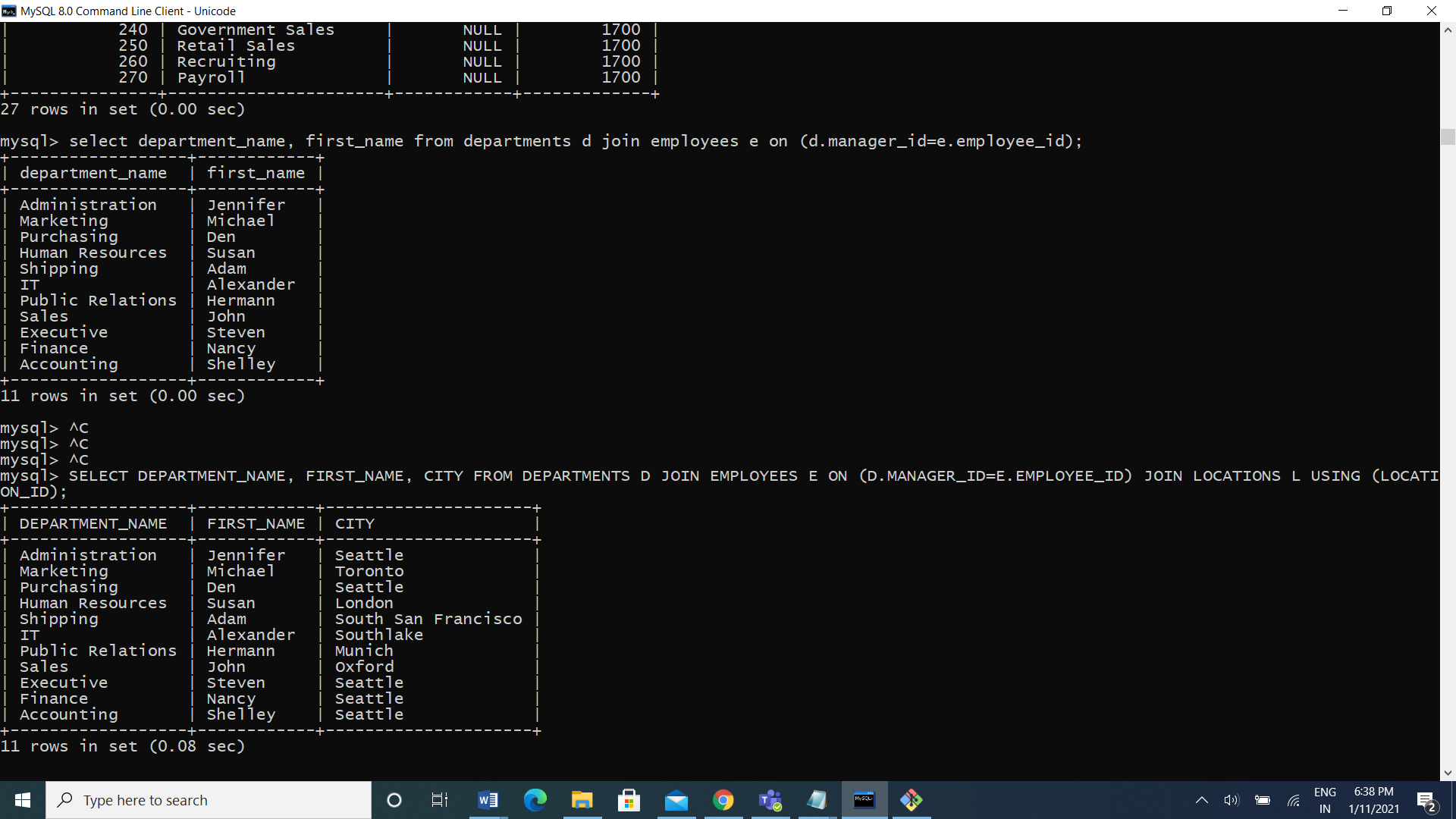
2) Display department name and manager first name.

Ans: select department\_name, first\_name from departments d join employees e on (d.manager\_id=e.employee\_id);

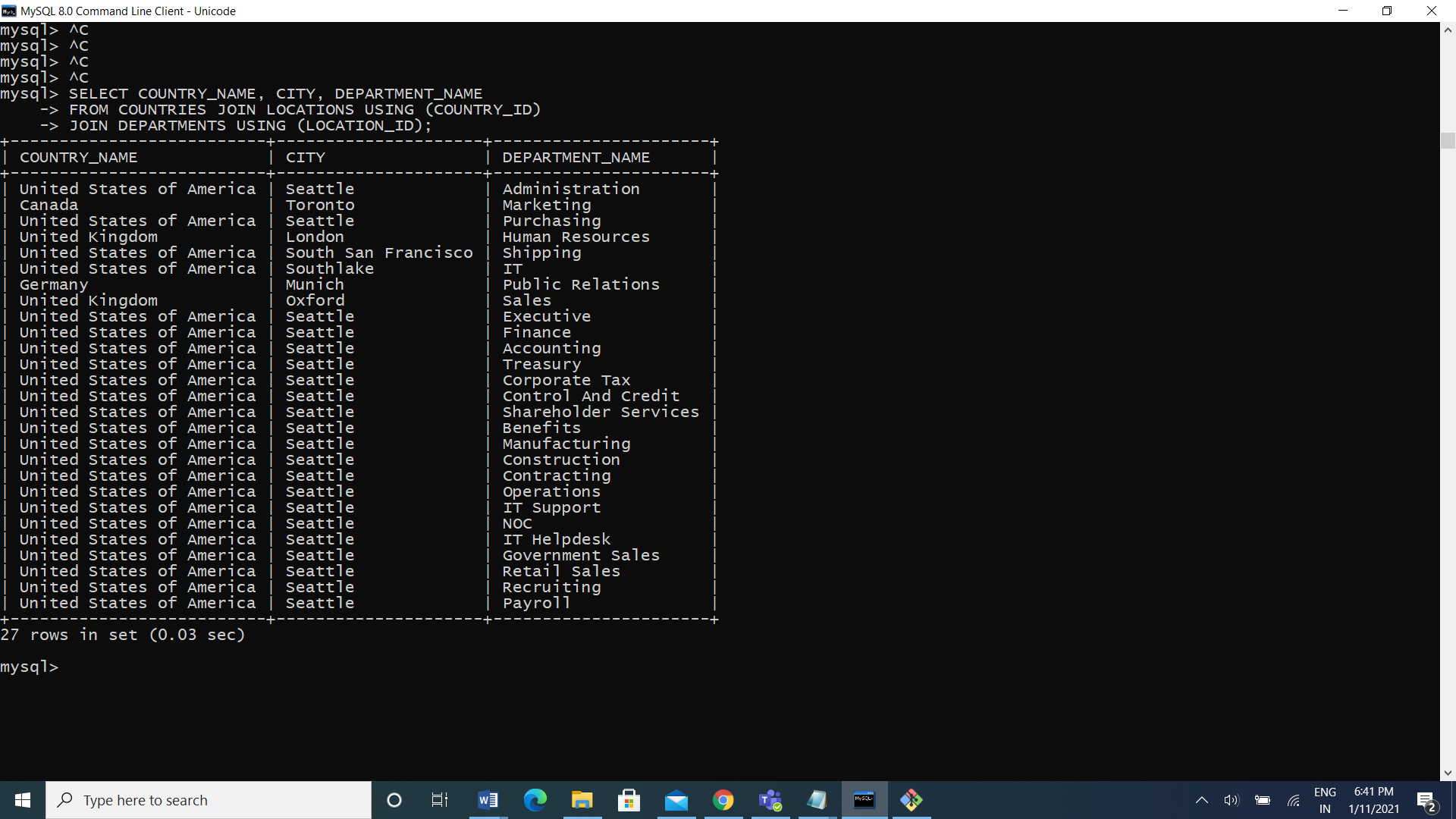


1. Display department name, manager name, and city.

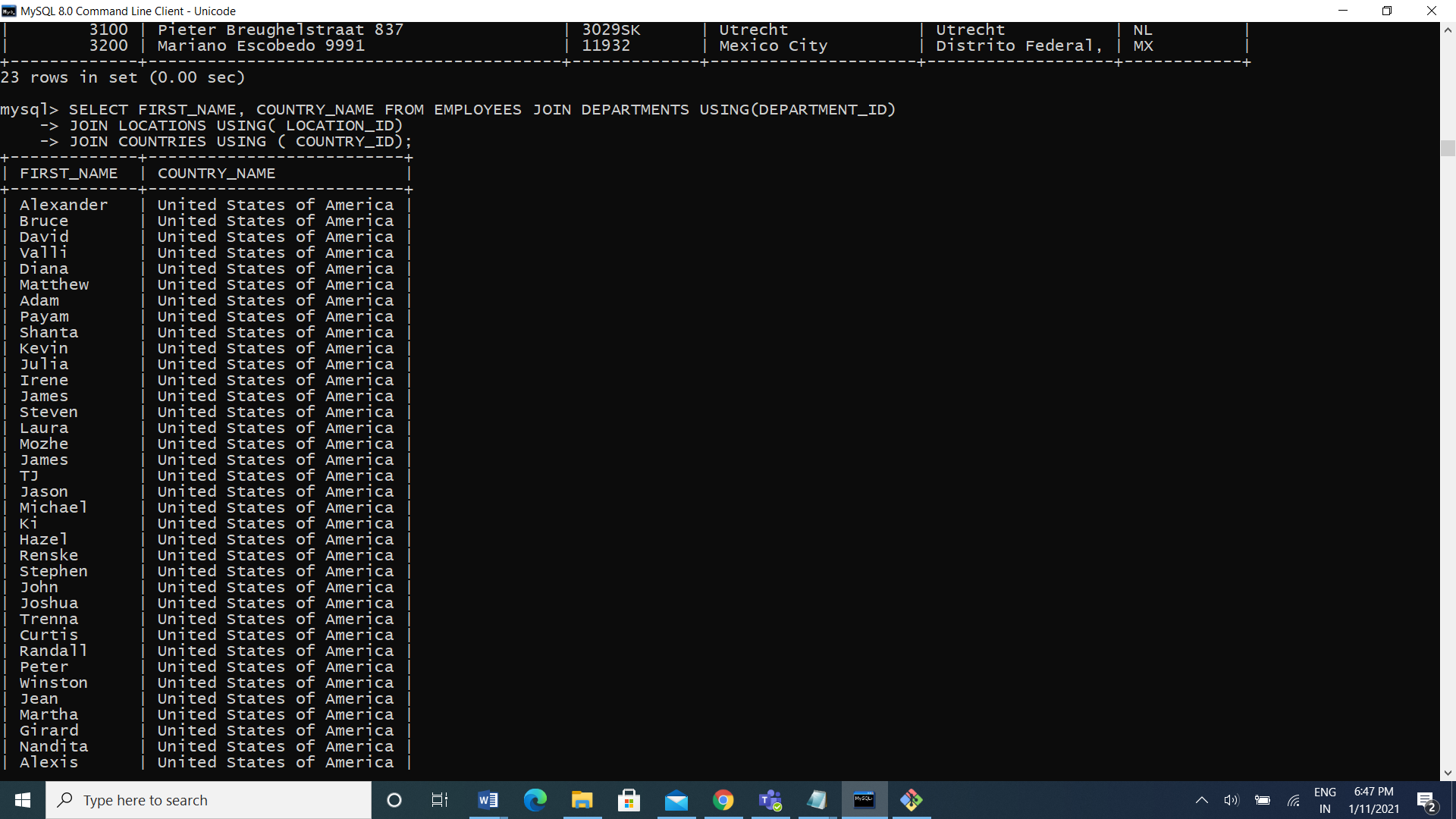
Ans: SELECT DEPARTMENT\_NAME, FIRST\_NAME, CITY FROM DEPARTMENTS D JOIN EMPLOYEES E ON (D.MANAGER\_ID=E.EMPLOYEE\_ID) JOIN LOCATIONS L USING (LOCATION\_ID);



1. Display country name, city, and department name.

Ans: select country\_name, city, department\_name from countries join location using(country\_id) join departments using(location\_id);

1. Display employee name and country in which he is working.

Ans: select first\_name, country\_name from employees join departments using(department\_id) join locations using(location\_id) join countries using(country\_id); 

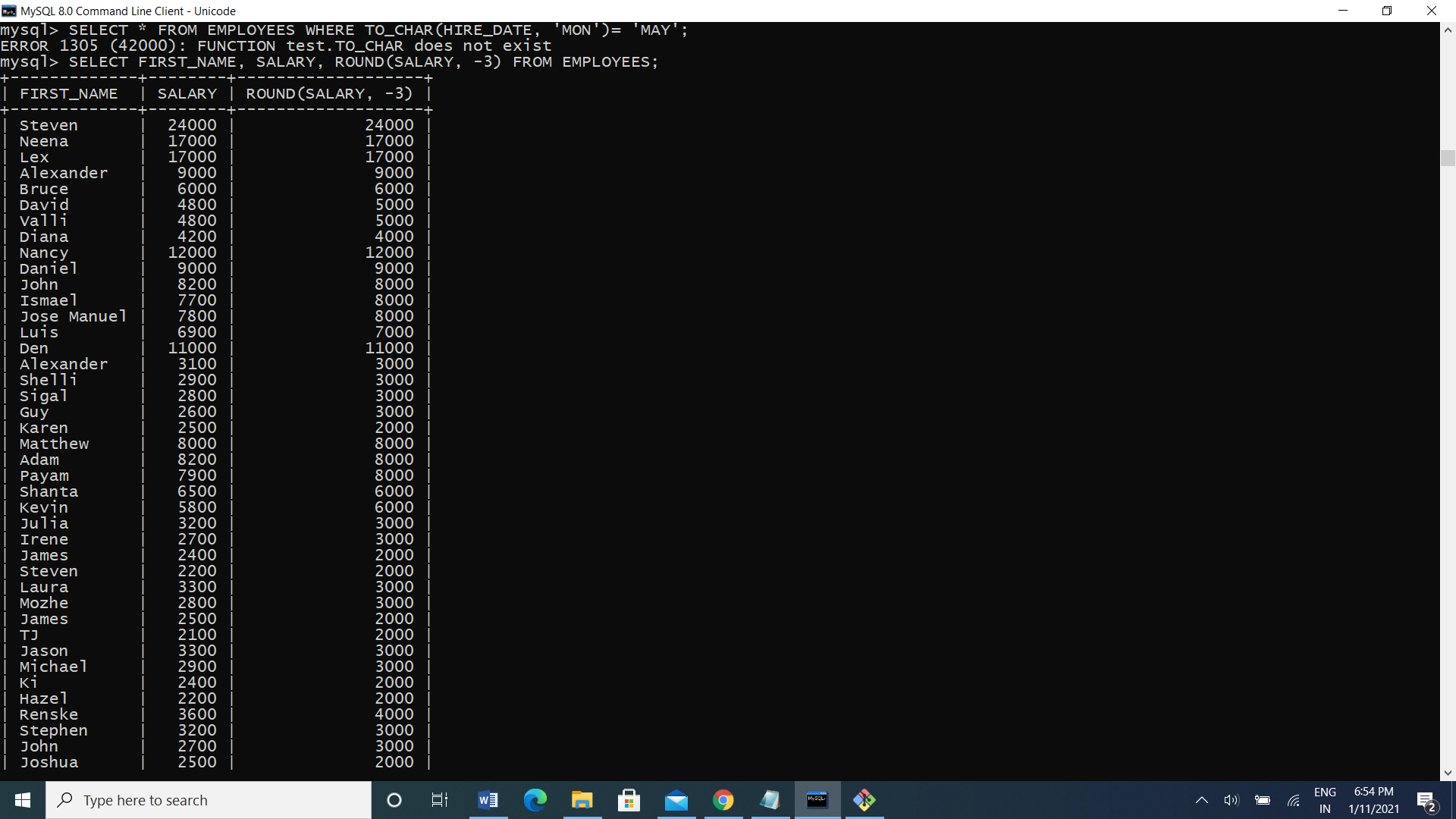
Functions

1. Display employees who joined in the month of May.

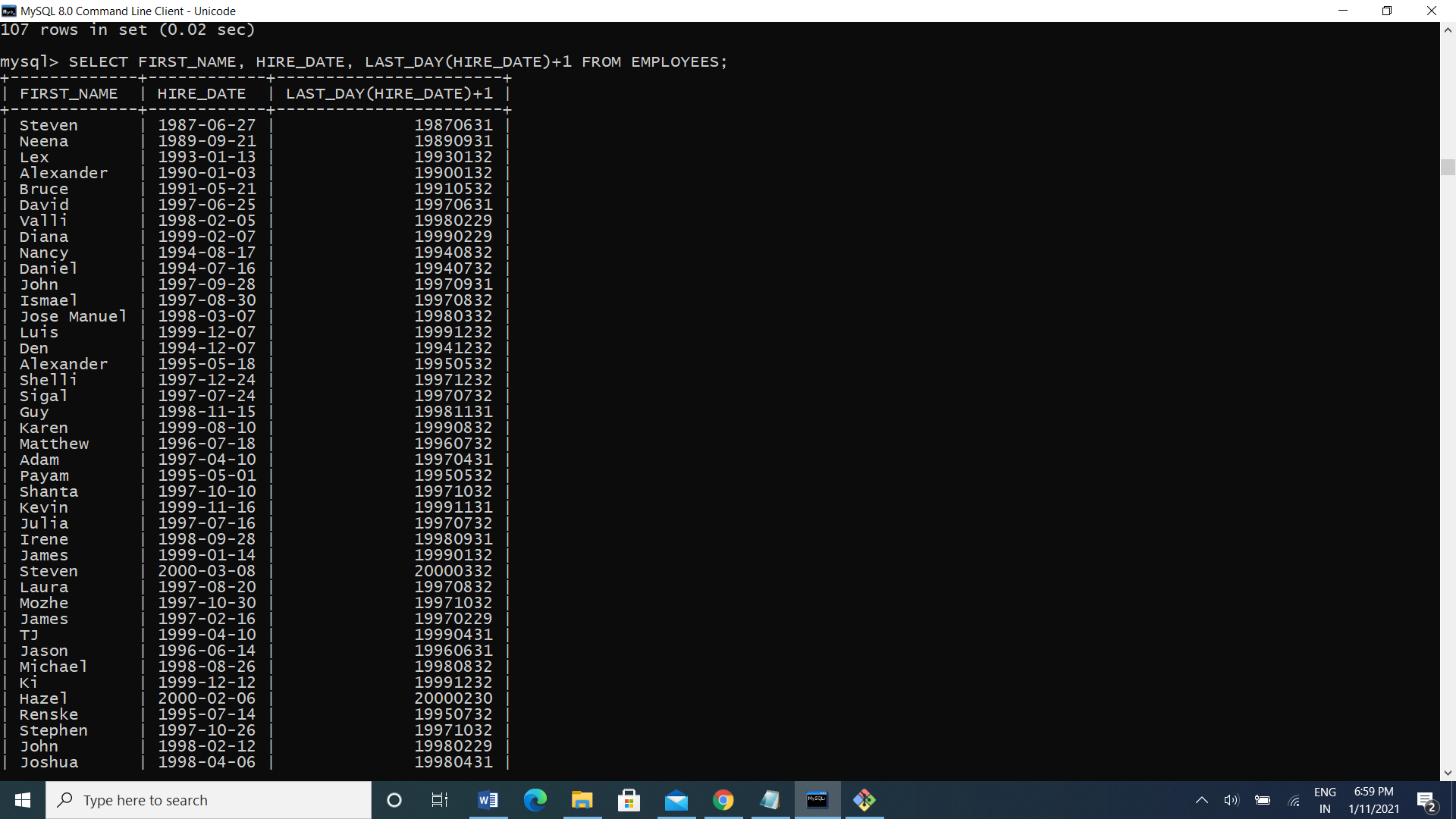
Ans: mysql> select first\_name,hire\_date from employees where hire\_date like '%05%';

2. Display first name, salary, and round the salary to thousands.

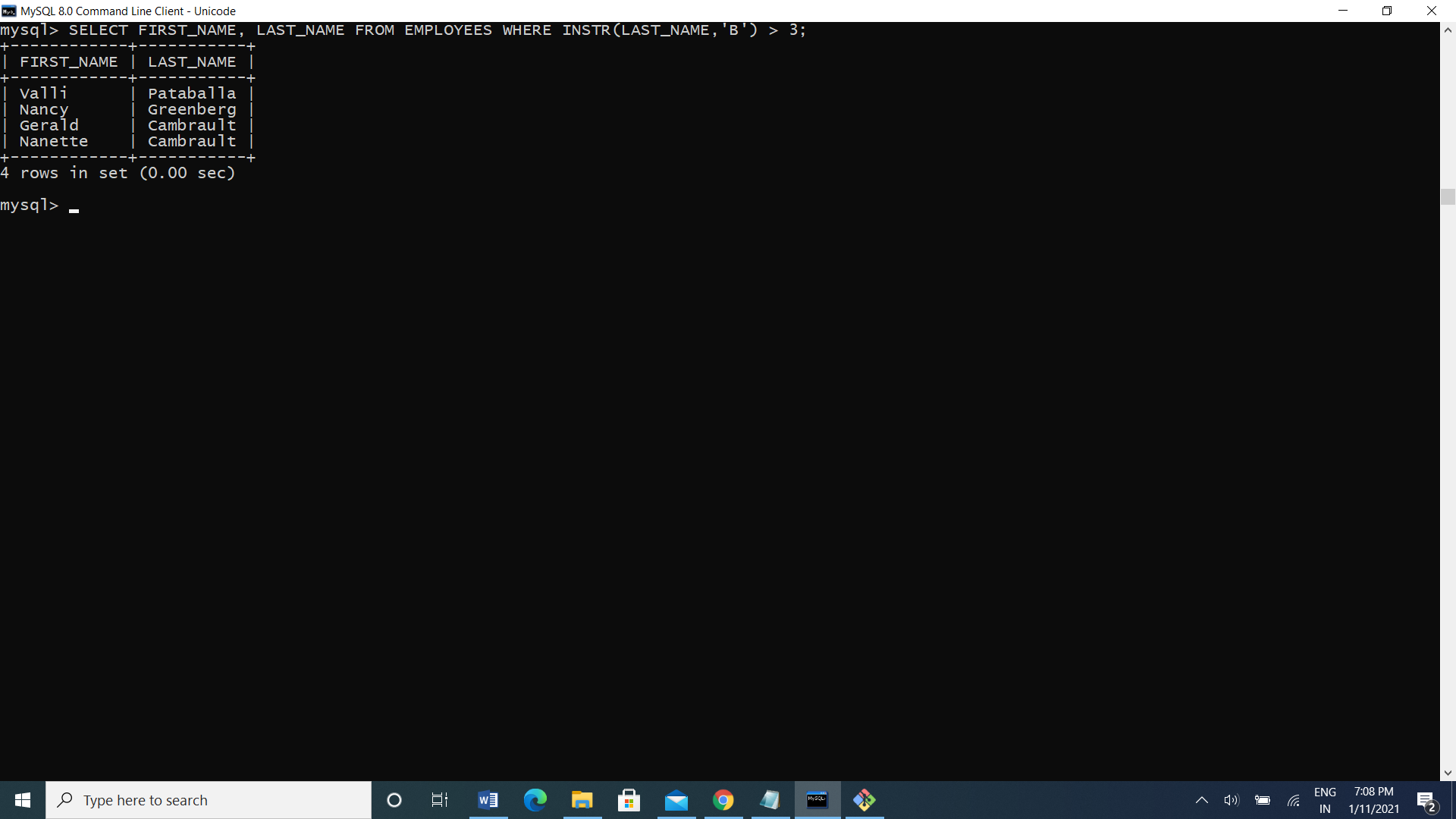
Ans: select first\_name, salary, round(salary, -3) from employees;



3. Display first name and date of first salary of the employees.

Ans: SELECT FIRST\_NAME, HIRE\_DATE, LAST\_DAY(HIRE\_DATE)+1 FROM EMPLOYEES;

4. Display first name and experience of the employees.

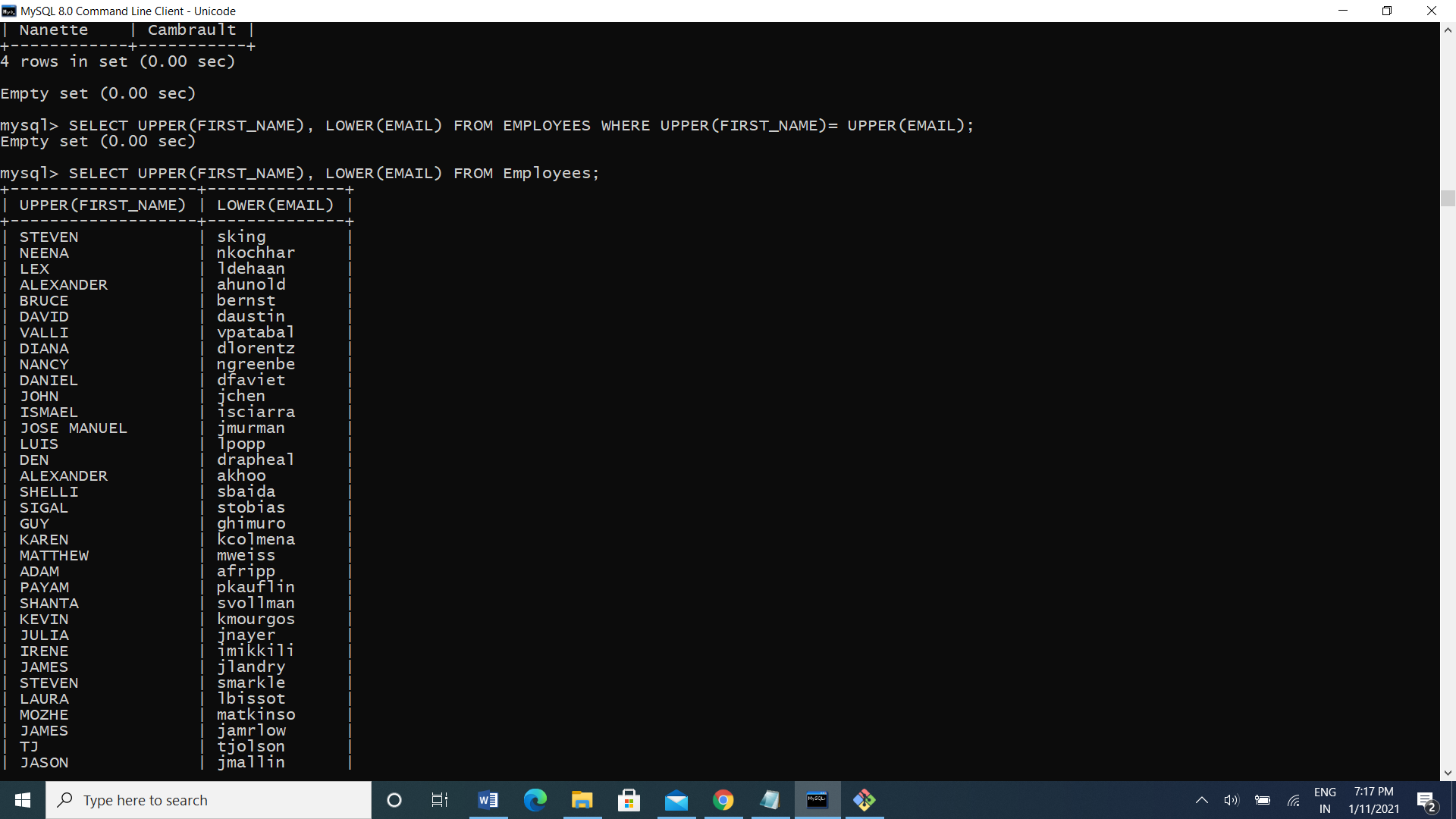
Ans: select first\_name, last\_name from employees where instr(last\_name, ‘B’)>3;

5. Display the length of first name for employees where last name contain character ‘b’ after 3rd position.

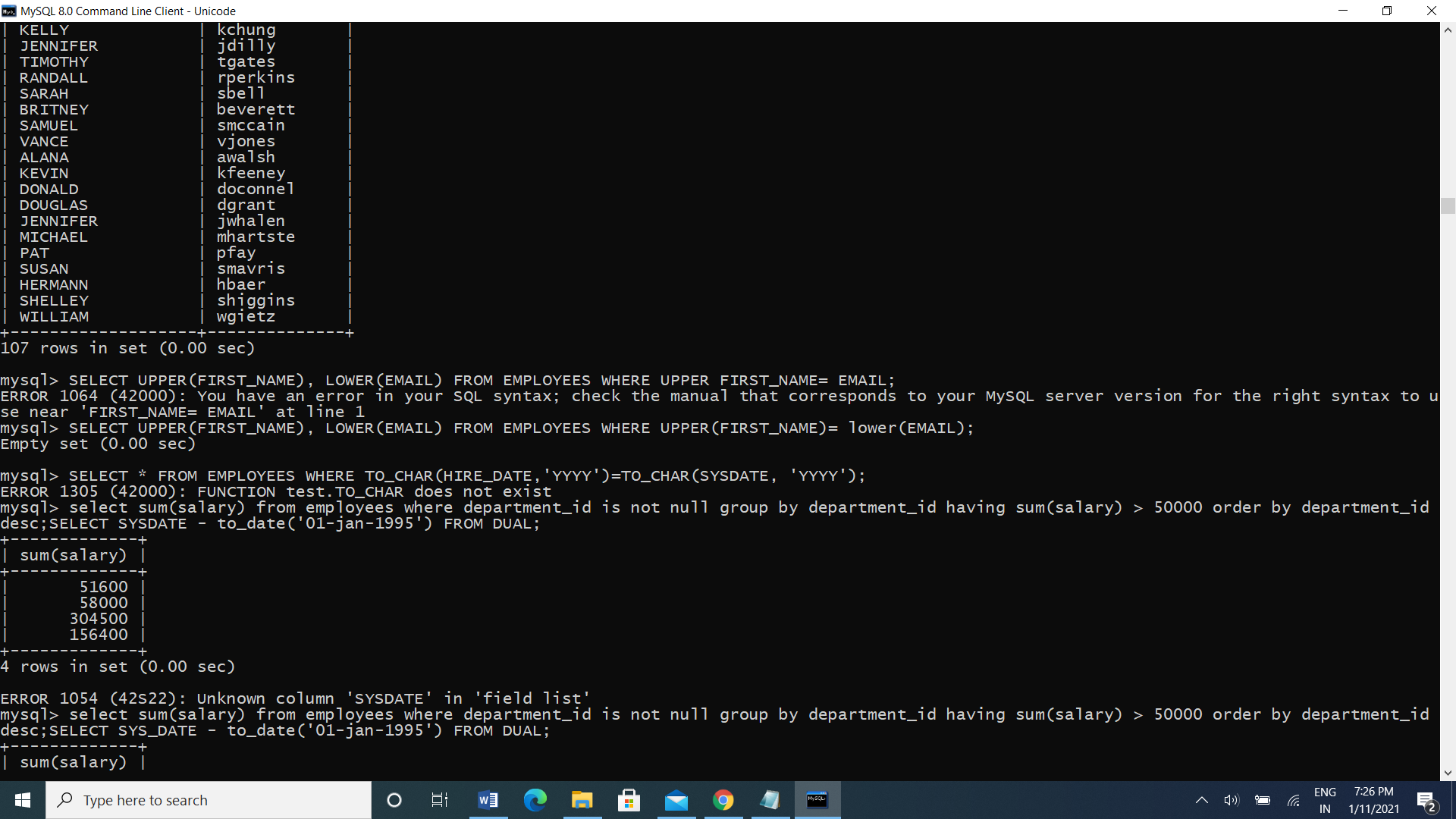
Ans: mysql> SELECT FIRST\_NAME, LAST\_NAME FROM EMPLOYEES WHERE INSTR(LAST\_NAME,'B') > 3;

6. Display first name in upper case and email address in lower case for employees where the first name and email address are same irrespective of the case.

Ans: select upper(first\_name), lower(email) where upper(first\_name)=upper(email);



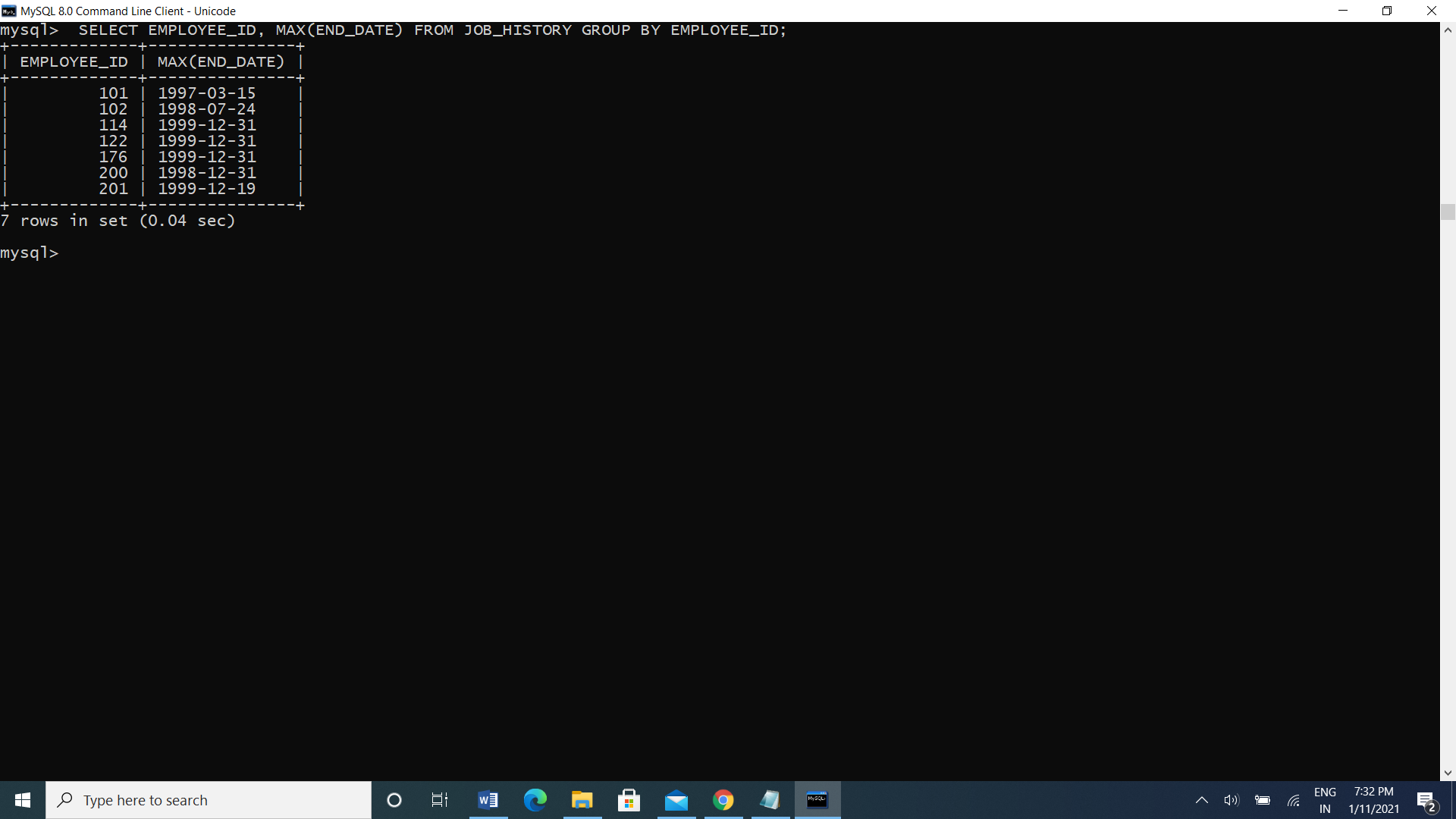
8. Display the number of days between system date and 1st January 1995.

Ans:select sysdate-to\_date(’01-jan-1995’)from dual;

MySQL Aggregate function

1. Display employee ID and the date on which he ended his previous job.

Ans: select employee\_id, max(end\_date) from job\_history f=group by employee\_id;



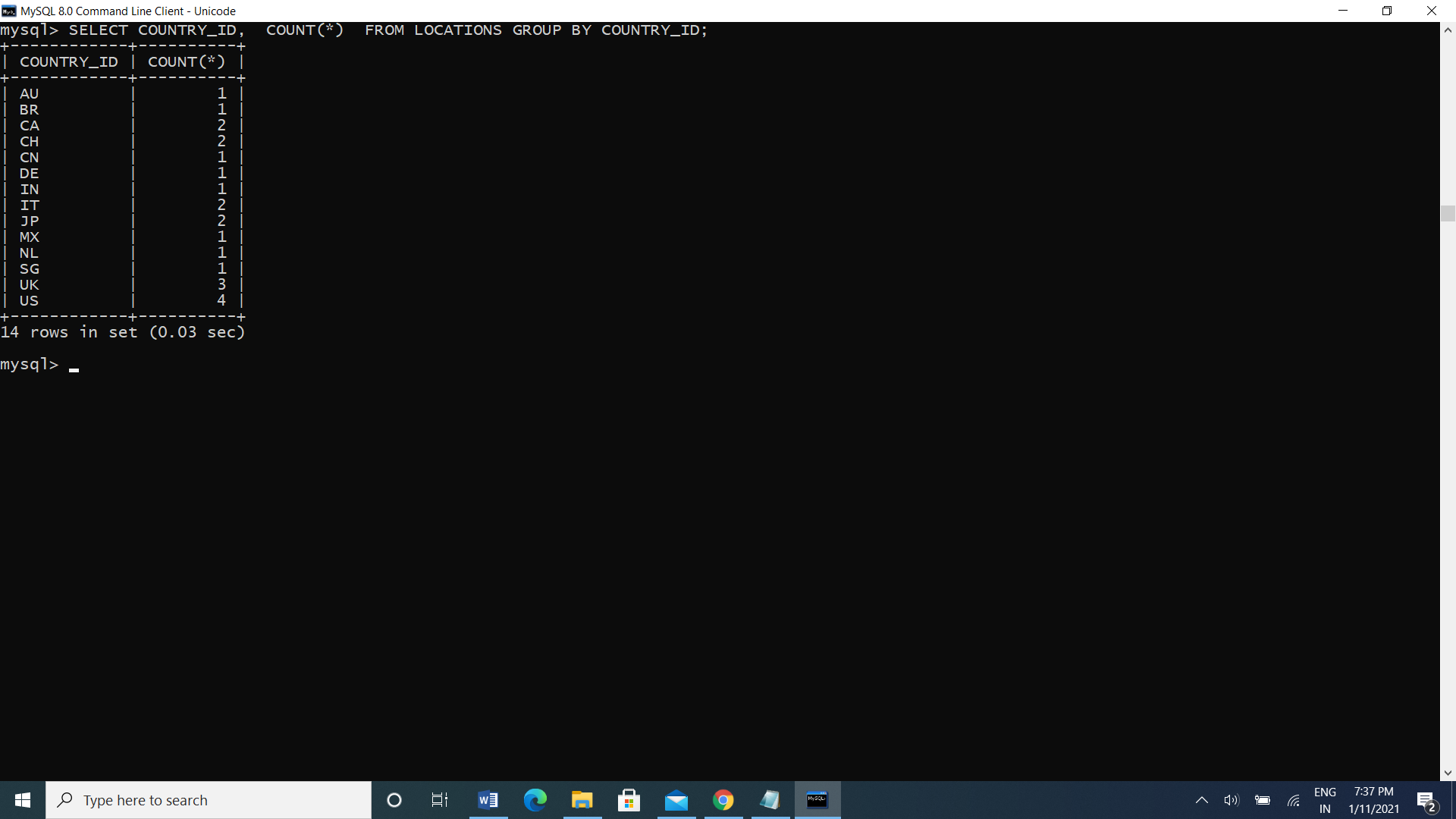
2.Display number of employees joined after 15th of the month.

Ans: mysql> SELECT COUNT(\*) FROM EMPLOYEES WHERE HIRE\_DATE>15;

select with Group by

1. Display the country ID and number of cities we have in the country.

Ans: SELECT COUNTRY\_ID, COUNT(\*) FROM LOCATIONS GROUP BY COUNTRY\_ID;

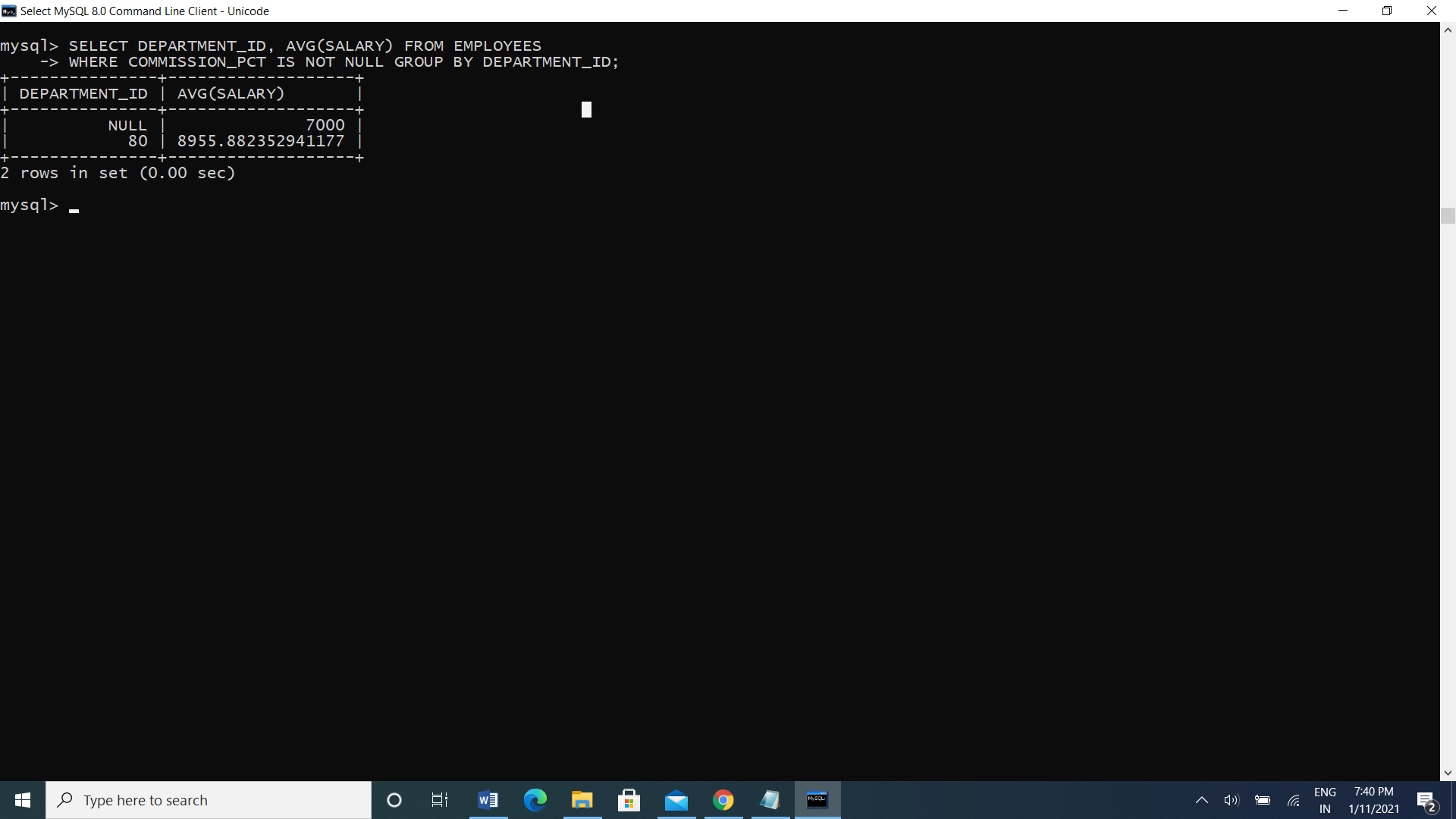


1. Display average salary of employees in each department who have commission percentage.

Ans: SELECT DEPARTMENT\_ID, AVG(SALARY) FROM EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL GROUP BY DEPARTMENT\_ID

3. Display job ID, number of employees, sum of salary, and difference between highest salary and lowest salary of the employees of the job.

Ans: SELECT JOB\_ID,COUNT(\*),SUM(SALARY),MAX(SALARY)-MIN(SALARY) SALARY FROM EMPLOYEES GROUP BY JOB\_ID;



6. Display job ID for jobs with average salary more than 10000.

Ans: mysql> SELECT JOB\_ID,AVG(SALARY) FROM EMPLOYEES GROUP BY JOB\_ID HAVING AVG(SALARY)>10000;

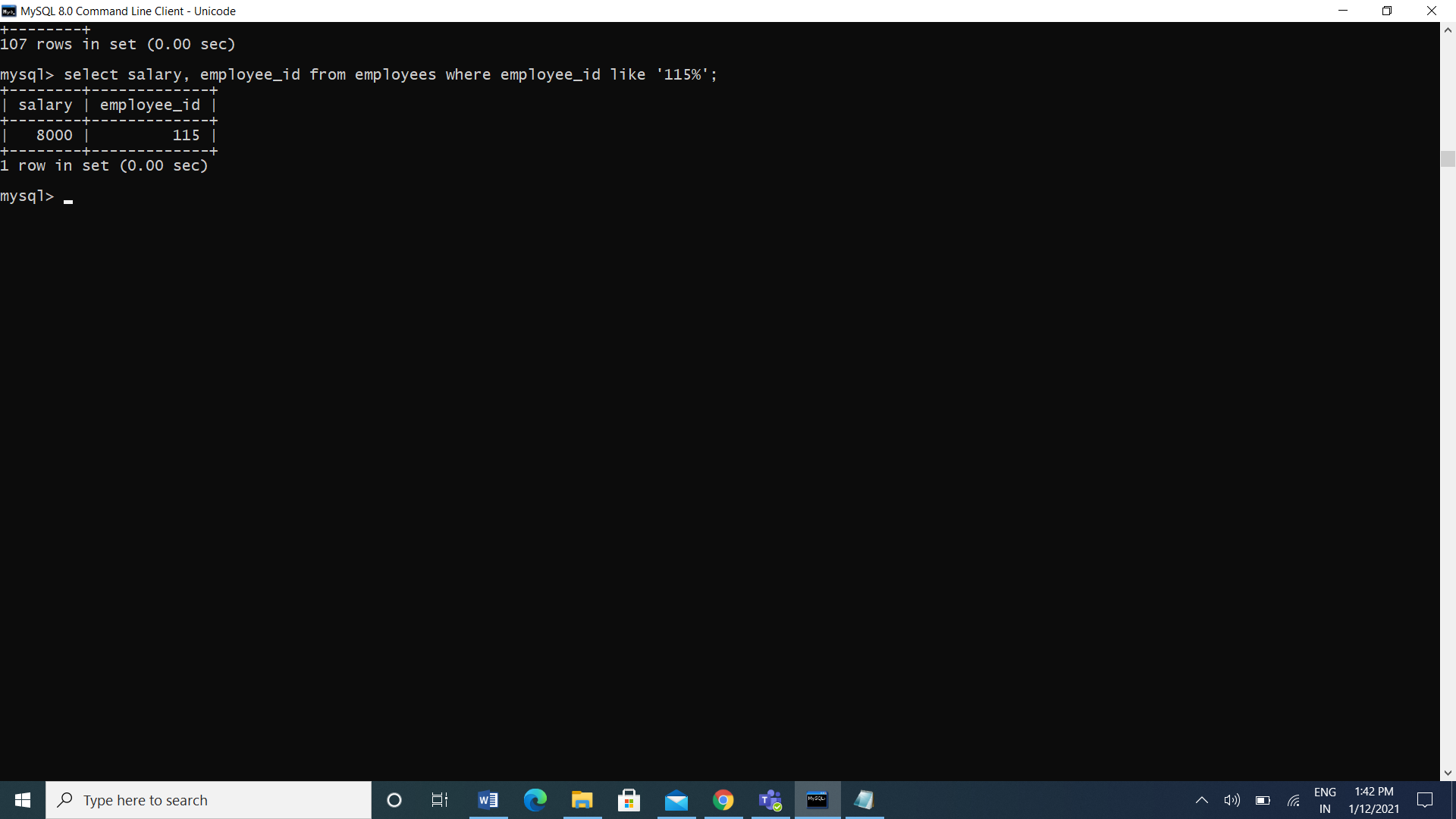
8. Display departments in which more than five employees have commission percentage.

Ans: mysql> select department\_id from employees where commission\_pct is not null group by job\_id HAVING COUNT(COMMISSION\_PCT)>5;

Day 2 (12-01-2021)

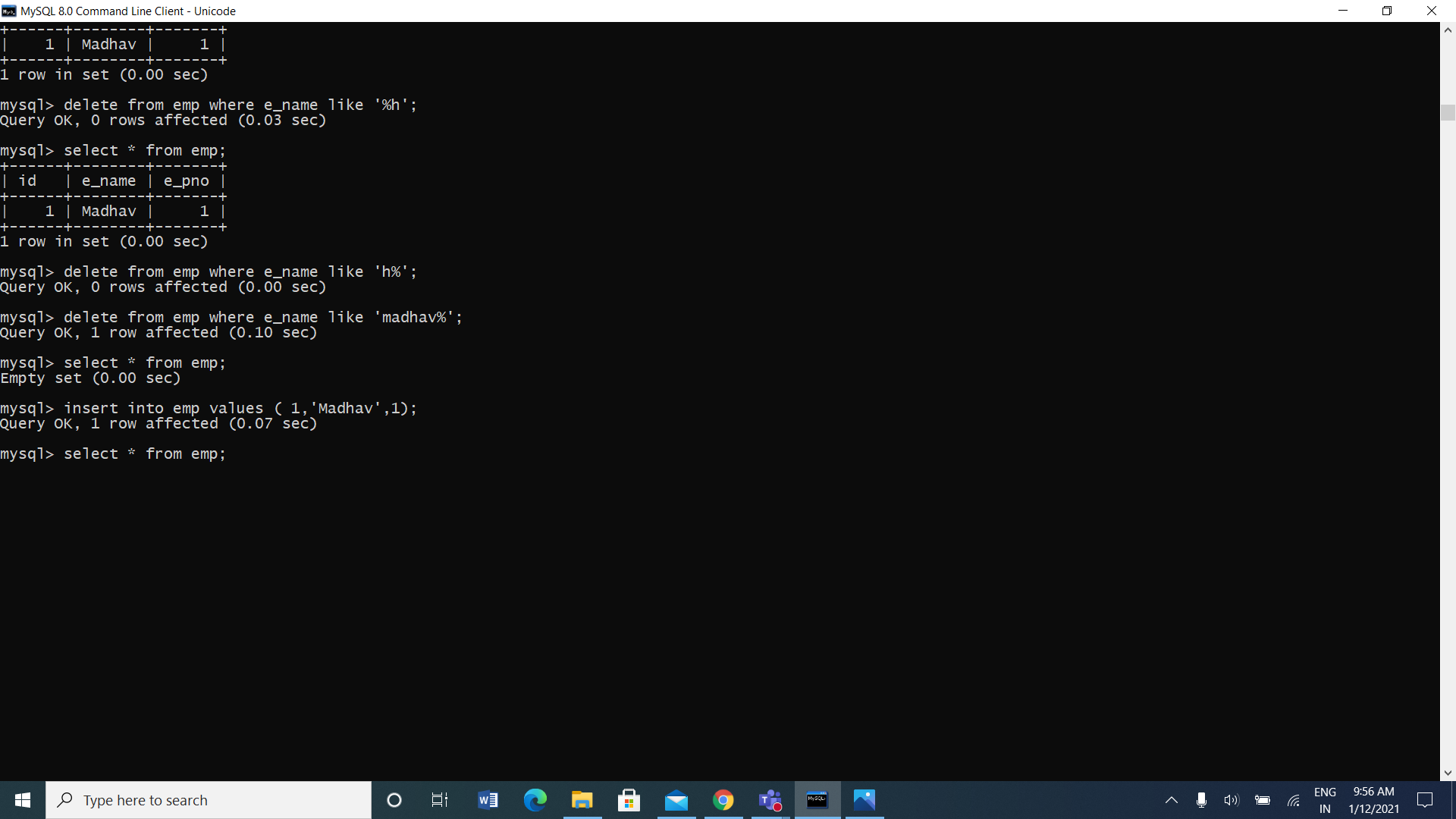
DML Operation:

1. Change salary of employee 115 to 8000 if the existing salary is less than 6000.

Ans: UPDATE EMPLOYEES SET SALARY = 8000 WHERE EMPLOYEE\_ID = 115 AND SALARY < 6000;

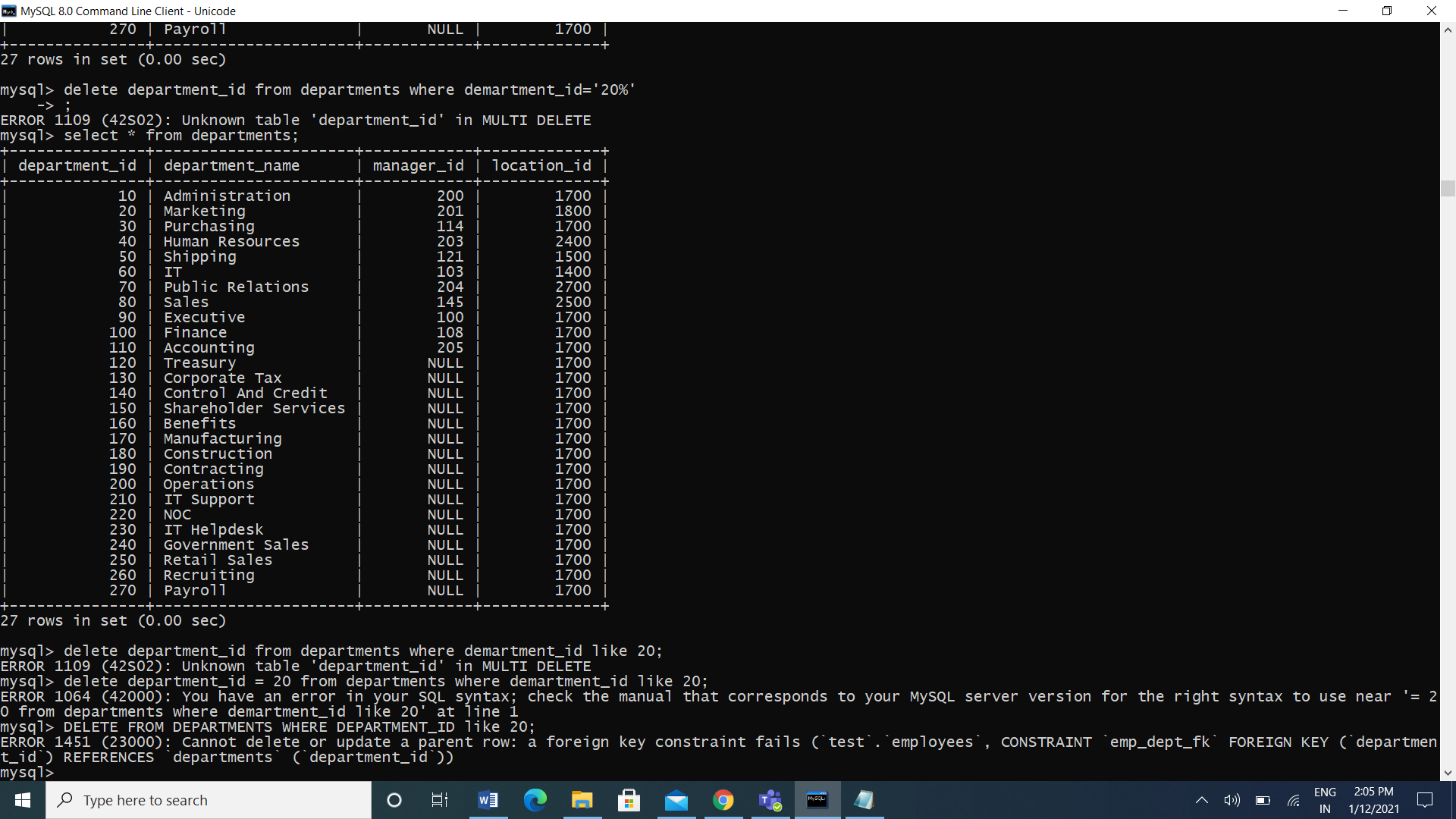
1. Insert a new employee into employees with all the required details.

Ans: insert into employees values(207, 'Madhav', 'Koti', 'madhav', null, SYSDATE(),'AC\_ACCOUNT', 25000, null, 115, 120);



1. Delete department 20.

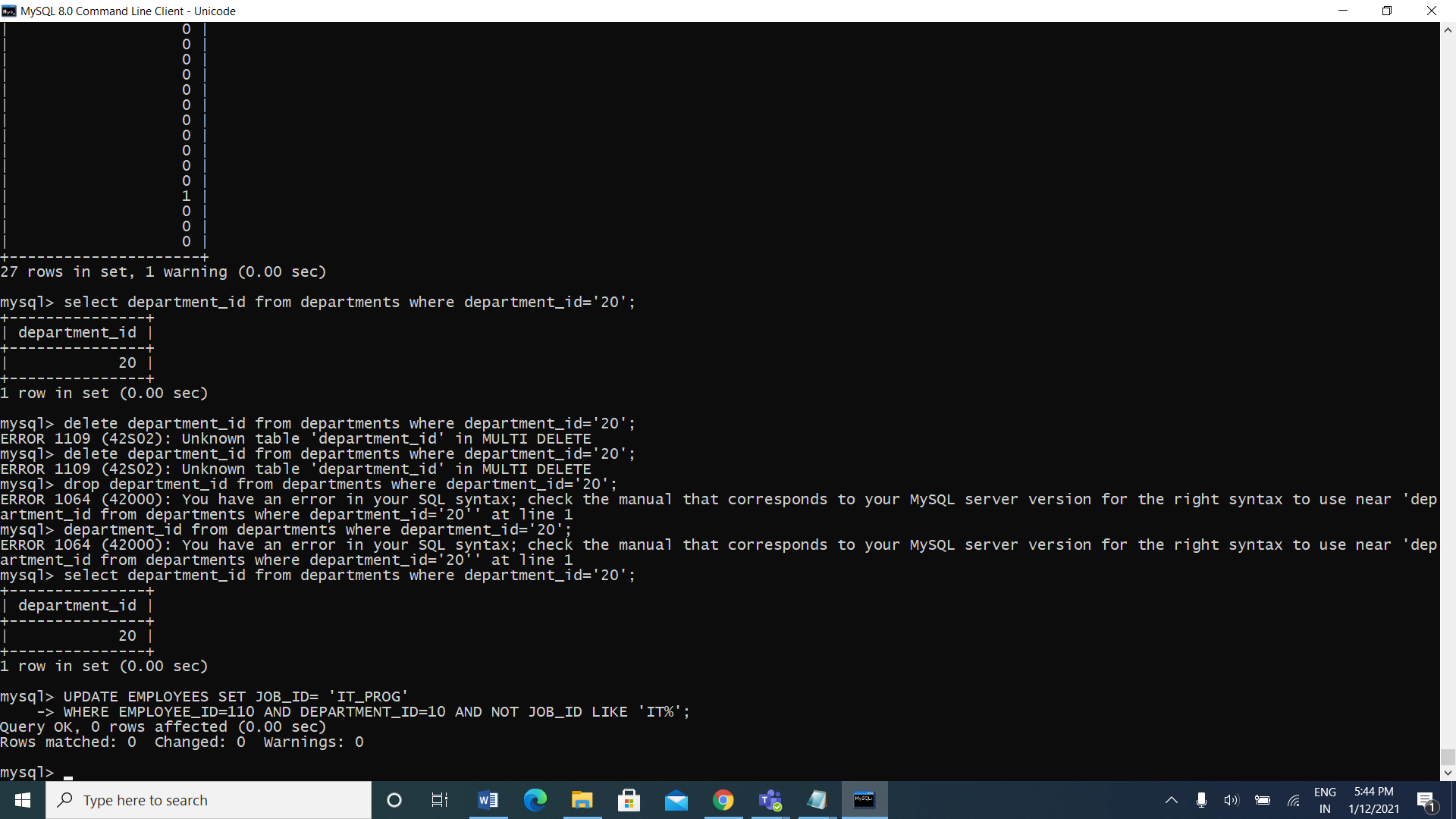
Ans: DELETE FROM DEPARTMENTS WHERE DEPARTMENT\_ID like 20;



1. Change job ID of employee 110 to IT\_PROG if the employee belongs to department 10 and the existing job ID does not start with IT.

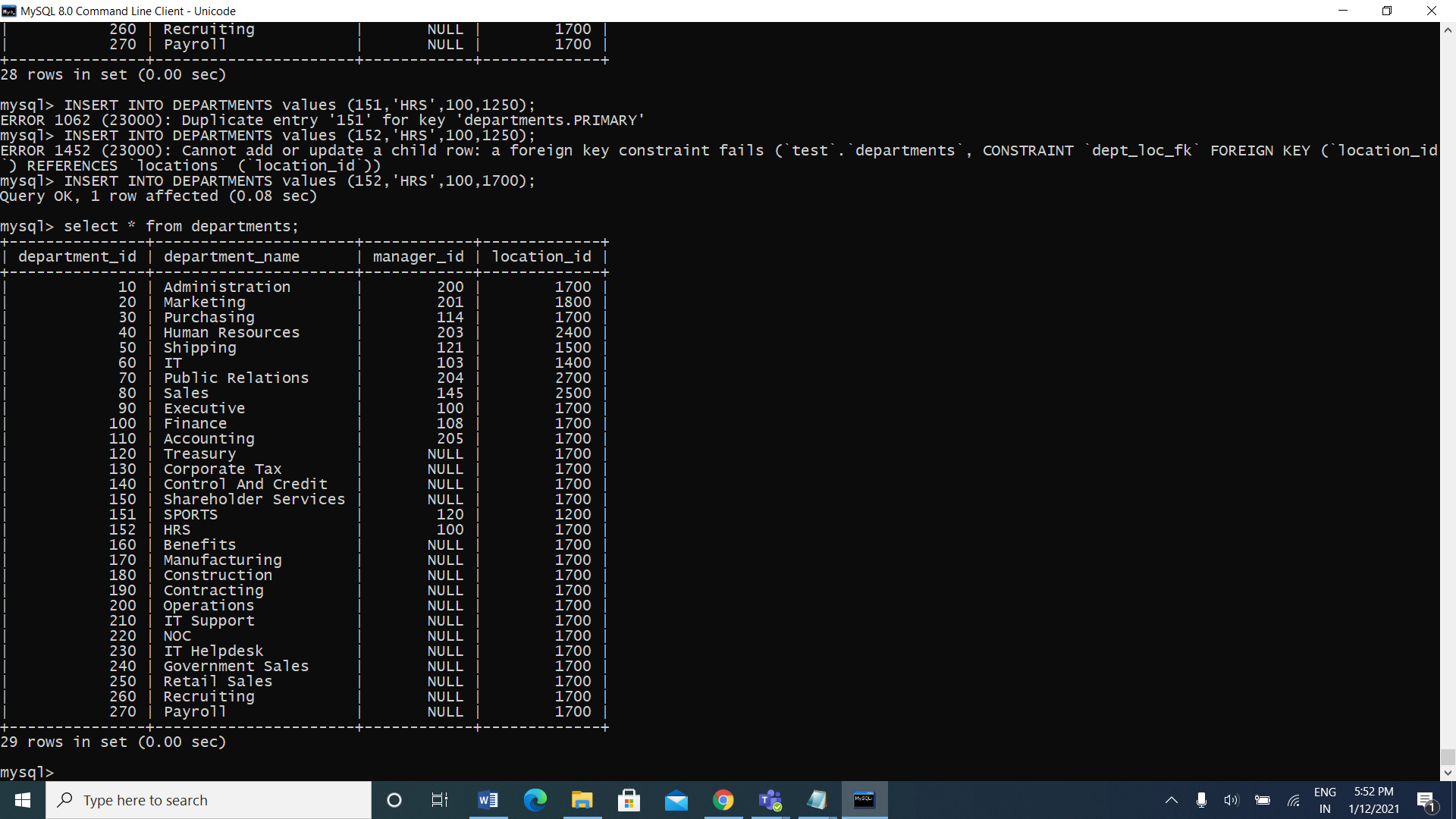
Ans: UPDATE EMPLOYEES SET JOB\_ID= 'IT\_PROG'

-> WHERE EMPLOYEE\_ID=110 AND DEPARTMENT\_ID=10 AND NOT JOB\_ID LIKE 'IT%' ;



5. Insert a row into departments table with manager ID 120 and location ID in any location ID for city Tokyo.

Ans: INSERT INTO DEPARTMENTS values (152,'HRS',100,1700);



1. Display job title, employee ID, number of days between ending date and starting date for all jobs in department 30 from job history.

Ans: SELECT EMPLOYEE\_ID, JOB\_TITLE, END\_DATE-START\_DATE DAYS

-> FROM JOB\_HISTORY NATURAL JOIN JOBS

-> WHERE DEPARTMENT\_ID=30;

DDL Assignments with Constraints

1. Table ---> Customer

custId, firstName,lastName,age,city, mobileNumber, dob

Add the Constraints

custId is Primary Key

firstName not null

age must be greater than 21

mobile must be unique

Ans: CREATE TABLE customer(custid int primary key, firstname varchar(255) not null, age int, city varchar(20), mobilenumber int unique,dob int, check(age>21));

1. Table ----> Branch

branchId, branchName, city

Add the Constraints

branchId is Primary Key

branchName not null

city not null

Ans: create table branch(branchID varchar(20) primary key, branchName varchar(20) not null, city varchar(20) not null);

1. Table -----> Account

accountNumber, openingBalance, typeOfAccount, status,BankId,CustId

Add the Constraints

accountNumber is primary key

openingBalance must be greater than 5000

typeOfAccount must be saving/current

BankId is foreign key refer to BranchId(Primary key) Branch table

CustId is foreign key refer to Customer(Primary key) Customer table

Ans: create table account(accountnumber int primary key, opening\_balance int, typeOfAccount enum('savings','current'), status varchar(10), BankId int, foreign key(BankId)references branch(branchID), CustId int, foreign key(CustId) references customer(custid), check(opening\_balance > 5000));

1. Table ----> Transaction

transactionId, transactionDate, MediumOfTransaction, TransactionAmount

Add the Constraints

transactionId is primary key

Ans: create table transaction(transactionid int primary key, transactiondate int, mediumoftransaction enum('online','ofline'), transactionamount int);

5.Table ----> Loan

LoanId, loanAmount, customerId and bankdId

Add the Constraints

loadId is primary key

loanAmount must be +ve

BankId is foreign key refer to BranchId(Primary key) Branch table

Ans: CREATE TABLE loan(custid VARCHAR(6),loan\_id varchar (10),bid VARCHAR(6),loan\_amount INT(7),CONSTRAINT loan\_customer\_custid\_bid\_pk PRIMARY KEY(custid,bid),CONSTRAINT loan\_custid\_fk FOREIGN KEY(custid) REFERENCES customer(custid),CONSTRAINT loan\_bid\_fk FOREIGN KEY(bid) REFERENCES branch(bid) );

Sub Query

1. Display details of departments managed by ‘John’.

Ans: SELECT department\_name FROM departments WHERE manager\_id IN (SELECT employee\_id FROM employees WHERE first\_name='john');

1. Display employees who did not do any job in the past.

Ans: select \* from employees where employee\_id not in(select employee\_id from job\_history);

1. Display job title and average salary for employees who did a job in the past.

Ans: SELECT JOB\_TITLE, AVG(SALARY) FROM JOBS, EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT EMPLOYEE\_ID FROM JOB\_HISTORY);

1. Display country name, city, and number of departments where department has more than 5 employees.

Ans: SELECT COUNTRY\_NAME, CITY, COUNT(DEPARTMENT\_ID) FROM COUNTRIES JOIN LOCATIONS USING (COUNTRY\_ID) JOIN DEPARTMENTS USING (LOCATION\_ID) WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING COUNT(DEPARTMENT\_ID)>5) GROUP BY COUNTRY\_NAME, CITY;

1. Display details of manager who manages more than 5 employees.

Ans: SELECT FIRST\_NAME FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT MANAGER\_ID FROM EMPLOYEES GROUP BY MANAGER\_ID HAVING COUNT(\*)>5);

6. Display details of current job for employees who worked as IT Programmers in the past.

Ans: mysql> select \* from jobs where job\_id in(select job\_id from employees where employee\_id in (select employee\_id from job\_history where job\_id='it\_prog'));

7. Display the details of employees drawing the highest salary in the department.

Ans: mysql> select \* from employees where salary=(select max(salary) from employees where department\_id=department\_id);

8. Display third highest salary of all employees

Ans: mysql> select salary from employees e where 2=(select count(distinct salary)from employees where salary >e.salary);