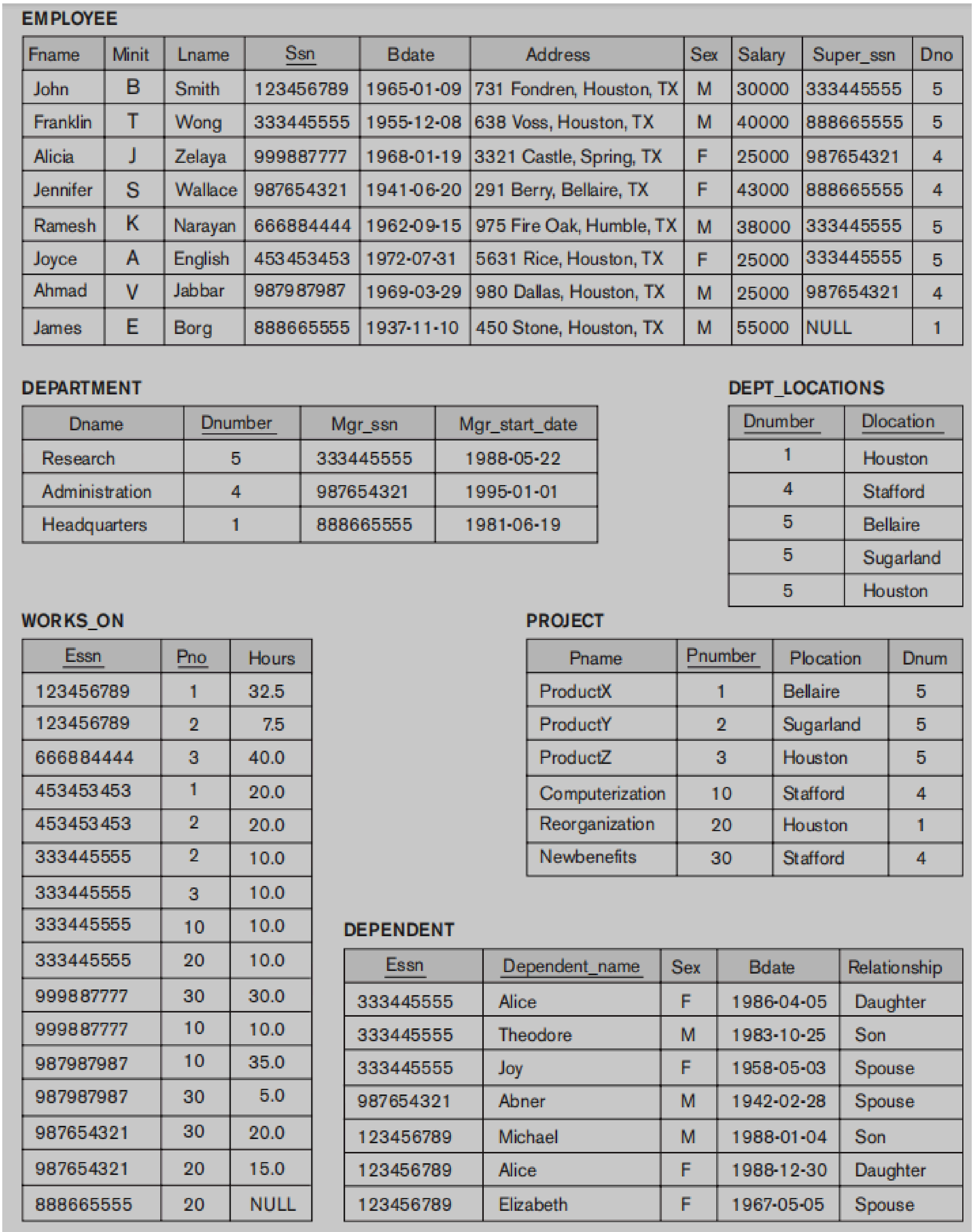
DataBase Management System
ASSIGNMENT 3

Name:Harsh Arora
Roll_No:AE-1218
Subject:DBMS
Semester:4th Semester
Date of Submission:30/03/2024
Course:BSc (hons)Computer Science
Acharya Narendra Dev College

Given by:Sharanjit Kaur Ma’am



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

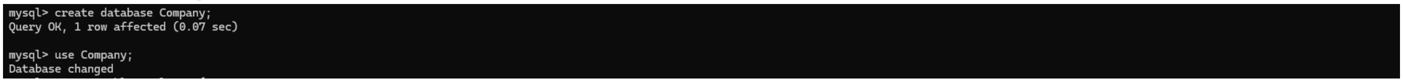
PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

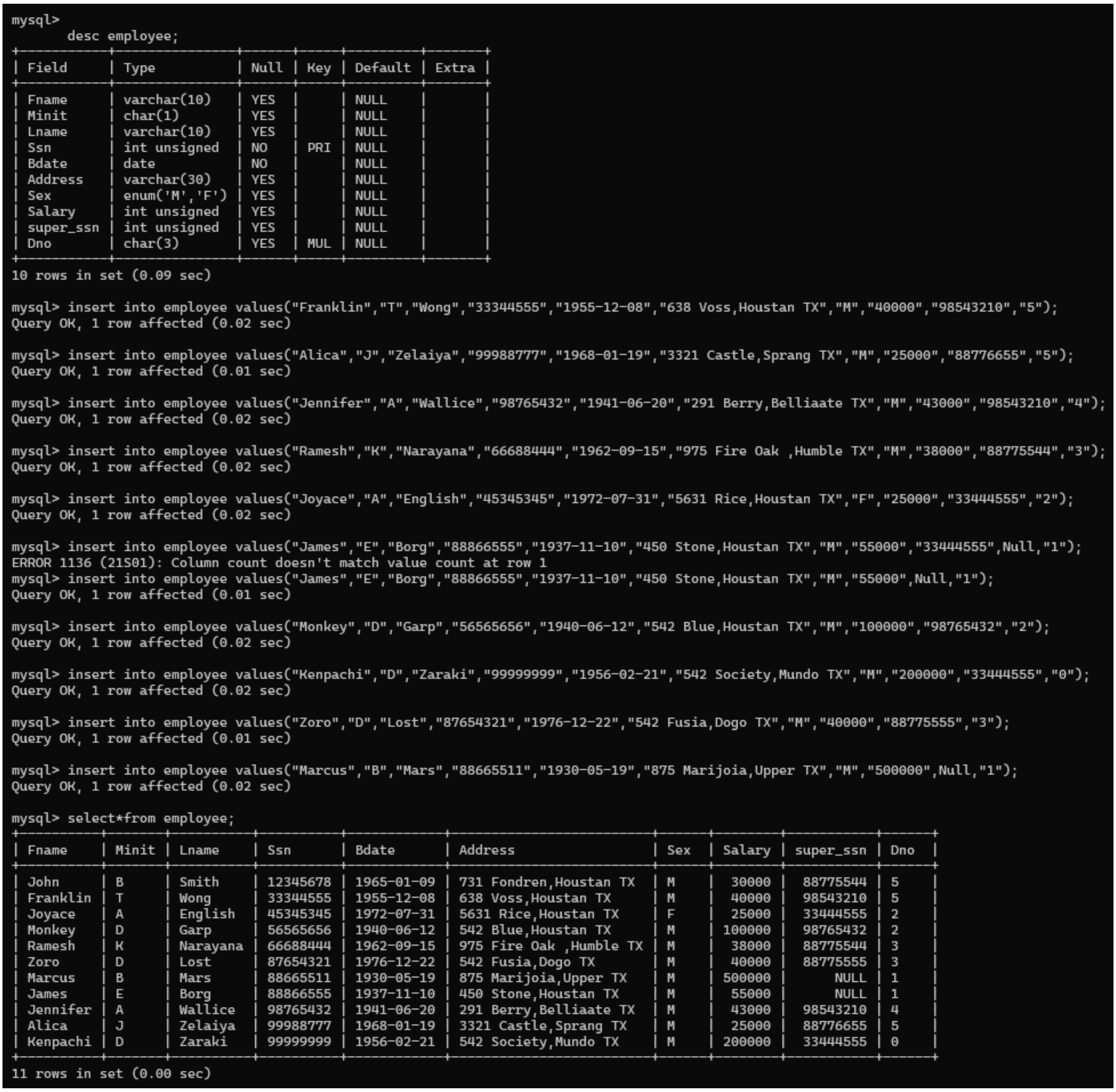
Creating DataBase:



```
mysql> create database Company;
Query OK, 1 row affected (0.07 sec)

mysql> use Company;
Database changed
```

Creating The Employee Table.

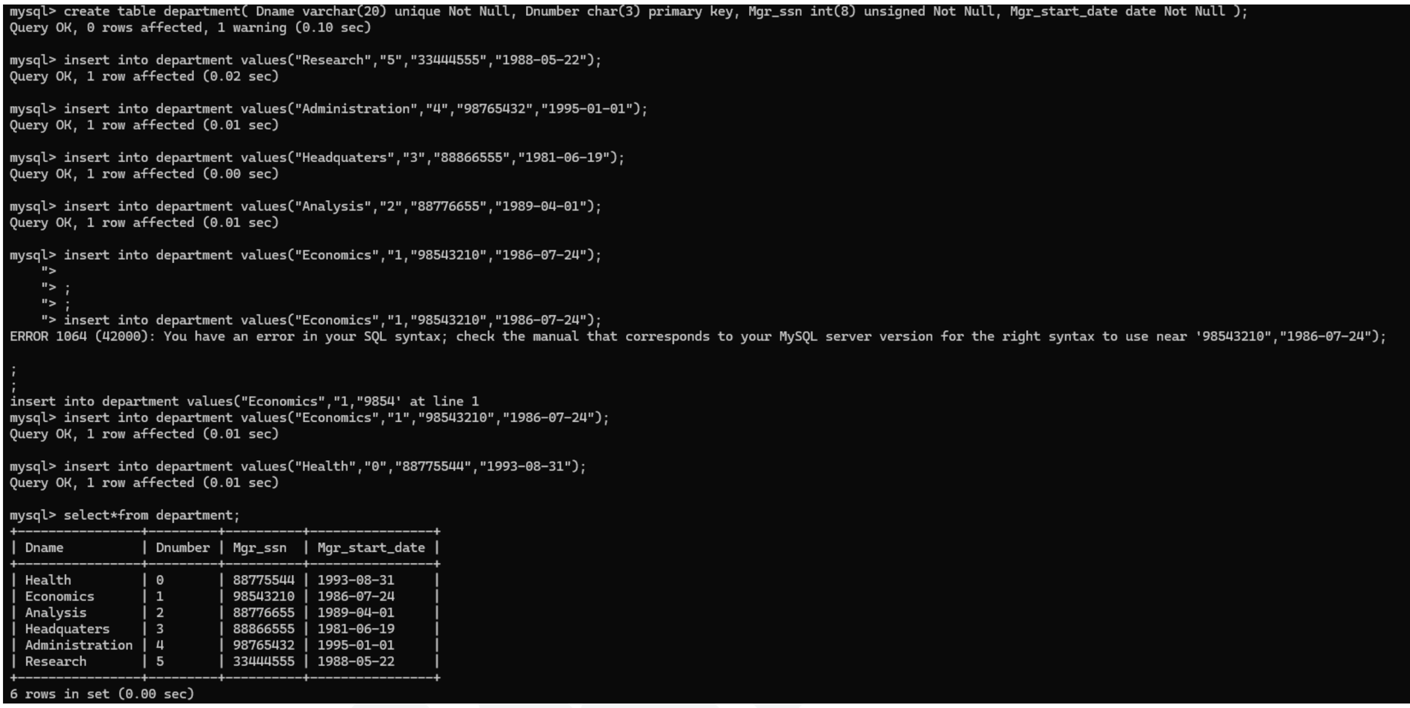
Inserting Values in it:



```
mysql>
       desc employee;
+-----------+---------------+------+-----+---------+-------+
| Field     | Type          | Null | Key | Default | Extra |
+-----------+---------------+------+-----+---------+-------+
| Fname     | varchar(10)   | YES  |     | NULL    |       |
| Minit     | char(1)       | YES  |     | NULL    |       |
| Lname     | varchar(10)   | YES  |     | NULL    |       |
| Ssn       | int unsigned  | NO   | PRI | NULL    |       |
| Bdate     | date          | NO   |     | NULL    |       |
| Address   | varchar(30)   | YES  |     | NULL    |       |
| Sex       | enum('M','F') | YES  |     | NULL    |       |
| Salary    | int unsigned  | YES  |     | NULL    |       |
| super_ssn | int unsigned  | YES  |     | NULL    |       |
| Dno       | char(3)       | YES  | MUL | NULL    |       |
+-----------+---------------+------+-----+---------+-------+
10 rows in set (0.09 sec)

mysql> insert into employee values("Franklin","T","Wong","33344555","1955-12-08","638 Voss,Houstan TX","M","40000","98543210","5");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("Alica","J","Zelaiya","99988777","1968-01-19","3321 Castle,Sprang TX","M","25000","88776655","5");
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values("Jennifer","A","Wallice","98765432","1941-06-20","291 Berry,Belliaate TX","M","43000","98543210","4");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("Ramesh","K","Narayana","66688444","1962-09-15","975 Fire Oak ,Humble TX","M","38000","88775544","3");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("Joyace","A","English","45345345","1972-07-31","5631 Rice,Houstan TX","F","25000","33444555","2");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("James","E","Borg","88866555","1937-11-10","450 Stone,Houstan TX","M","55000","33444555",Null,"1");
ERROR 1136 (21S01): Column count doesn't match value count at row 1
mysql> insert into employee values("James","E","Borg","88866555","1937-11-10","450 Stone,Houstan TX","M","55000",Null,"1");
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values("Monkey","D","Garp","56565656","1940-06-12","542 Blue,Houstan TX","M","100000","98765432","2");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("Kenpachi","D","Zaraki","99999999","1956-02-21","542 Society,Mundo TX","M","200000","33444555","0");
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values("Zoro","D","Lost","87654321","1976-12-22","542 Fusia,Dogo TX","M","40000","88775555","3");
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values("Marcus","B","Mars","88665511","1930-05-19","875 Marijoia,Upper TX","M","500000",Null,"1");
Query OK, 1 row affected (0.02 sec)

mysql> select*from employee;
+----------+-------+----------+----------+------------+-------------------------+-----+--------+-----------+------+
| Fname    | Minit | Lname    | Ssn      | Bdate      | Address                 | Sex | Salary | super_ssn | Dno  |
+----------+-------+----------+----------+------------+-------------------------+-----+--------+-----------+------+
| John     | B     | Smith    | 12345678 | 1965-01-09 | 731 Fondren,Houstan TX  | M   |  30000 |  88775544 | 5    |
| Franklin | T     | Wong     | 33344555 | 1955-12-08 | 638 Voss,Houstan TX     | M   |  40000 |  98543210 | 5    |
| Joyace   | A     | English  | 45345345 | 1972-07-31 | 5631 Rice,Houstan TX    | F   |  25000 |  33444555 | 2    |
| Monkey   | D     | Garp     | 56565656 | 1940-06-12 | 542 Blue,Houstan TX     | M   | 100000 |  98765432 | 2    |
| Ramesh   | K     | Narayana | 66688444 | 1962-09-15 | 975 Fire Oak ,Humble TX | M   |  38000 |  88775544 | 3    |
| Zoro     | D     | Lost     | 87654321 | 1976-12-22 | 542 Fusia,Dogo TX       | M   |  40000 |  88775555 | 3    |
| Marcus   | B     | Mars     | 88665511 | 1930-05-19 | 875 Marijoia,Upper TX   | M   | 500000 |      NULL | 1    |
| James    | E     | Borg     | 88866555 | 1937-11-10 | 450 Stone,Houstan TX    | M   |  55000 |      NULL | 1    |
| Jennifer | A     | Wallice  | 98765432 | 1941-06-20 | 291 Berry,Belliaate TX  | M   |  43000 |  98543210 | 4    |
| Alica    | J     | Zelaiya  | 99988777 | 1968-01-19 | 3321 Castle,Sprang TX   | M   |  25000 |  88776655 | 5    |
| Kenpachi | D     | Zaraki   | 99999999 | 1956-02-21 | 542 Society,Mundo TX    | M   | 200000 |  33444555 | 0    |
+----------+-------+----------+----------+------------+-------------------------+-----+--------+-----------+------+
11 rows in set (0.00 sec)
```

Creating TheDepartment Table.

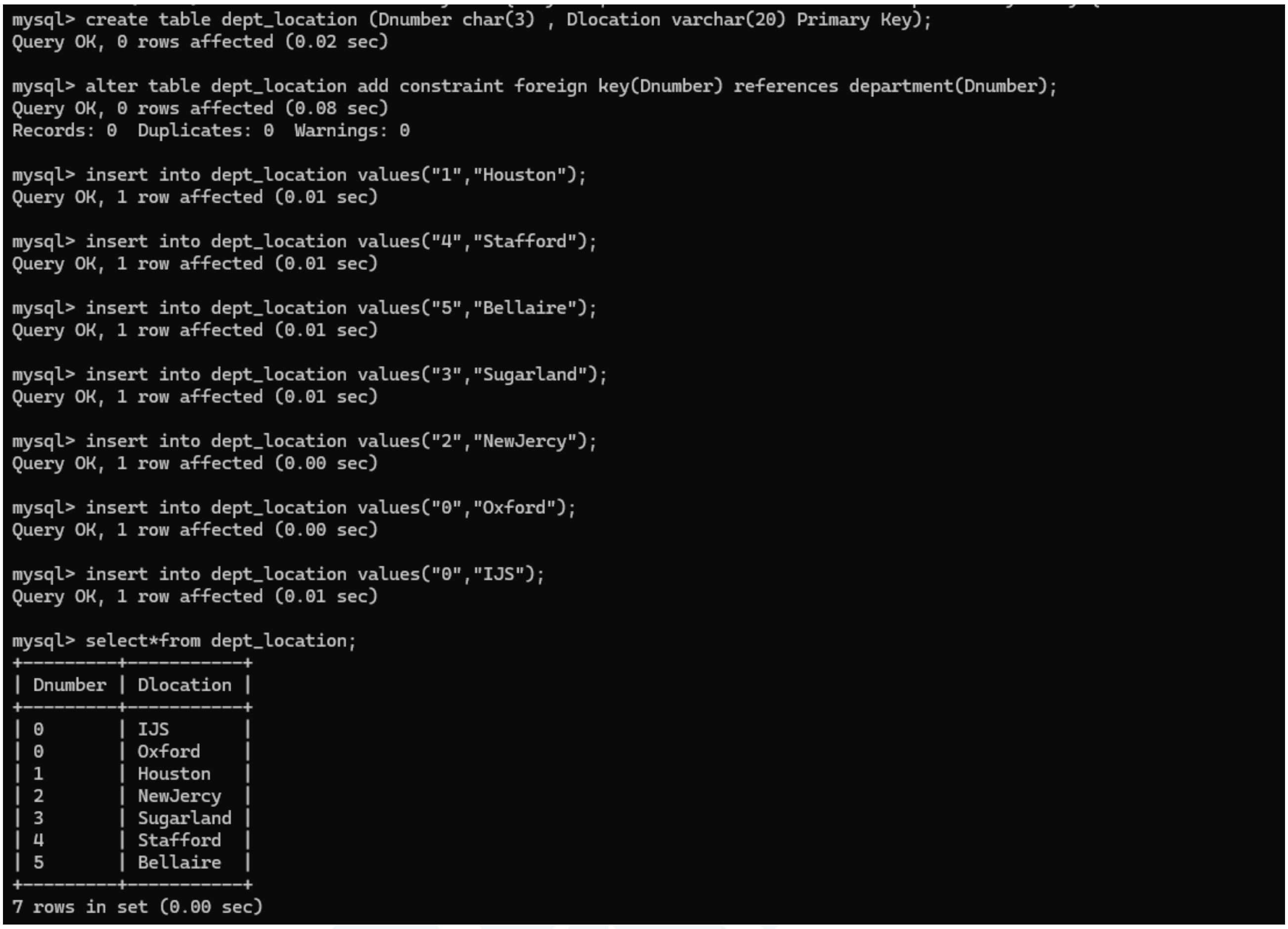
Inserting Values in it:



```
mysql> create table department( Dname varchar(20) unique Not Null, Dnumber char(3) primary key, Mgr_ssn int(8) unsigned Not Null, Mgr_start_date date Not Null );
Query OK, 0 rows affected, 1 warning (0.10 sec)

mysql> insert into department values("Research","5","33444555","1988-05-22");
Query OK, 1 row affected (0.02 sec)

mysql> insert into department values("Administration","4","98765432","1995-01-01");
Query OK, 1 row affected (0.01 sec)

mysql> insert into department values("Headquaters","3","88866555","1981-06-19");
Query OK, 1 row affected (0.00 sec)

mysql> insert into department values("Analysis","2","88776655","1989-04-01");
Query OK, 1 row affected (0.01 sec)

mysql> insert into department values("Economics","1,"98543210","1986-07-24");
    ">
    "> ;
    "> ;
    "> insert into department values("Economics","1,"98543210","1986-07-24");
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '98543210","1986-07-24");

;
;
insert into department values("Economics","1,"9854' at line 1
mysql> insert into department values("Economics","1","98543210","1986-07-24");
Query OK, 1 row affected (0.01 sec)

mysql> insert into department values("Health","0","88775544","1993-08-31");
Query OK, 1 row affected (0.01 sec)

mysql> select*from department;
+----------------+---------+----------+----------------+
| Dname          | Dnumber | Mgr_ssn  | Mgr_start_date |
+----------------+---------+----------+----------------+
| Health         | 0       | 88775544 | 1993-08-31     |
| Economics      | 1       | 98543210 | 1986-07-24     |
| Analysis       | 2       | 88776655 | 1989-04-01     |
| Headquaters    | 3       | 88866555 | 1981-06-19     |
| Administration | 4       | 98765432 | 1995-01-01     |
| Research       | 5       | 33444555 | 1988-05-22     |
+----------------+---------+----------+----------------+
6 rows in set (0.00 sec)
```

Creating The Department_locations Table.

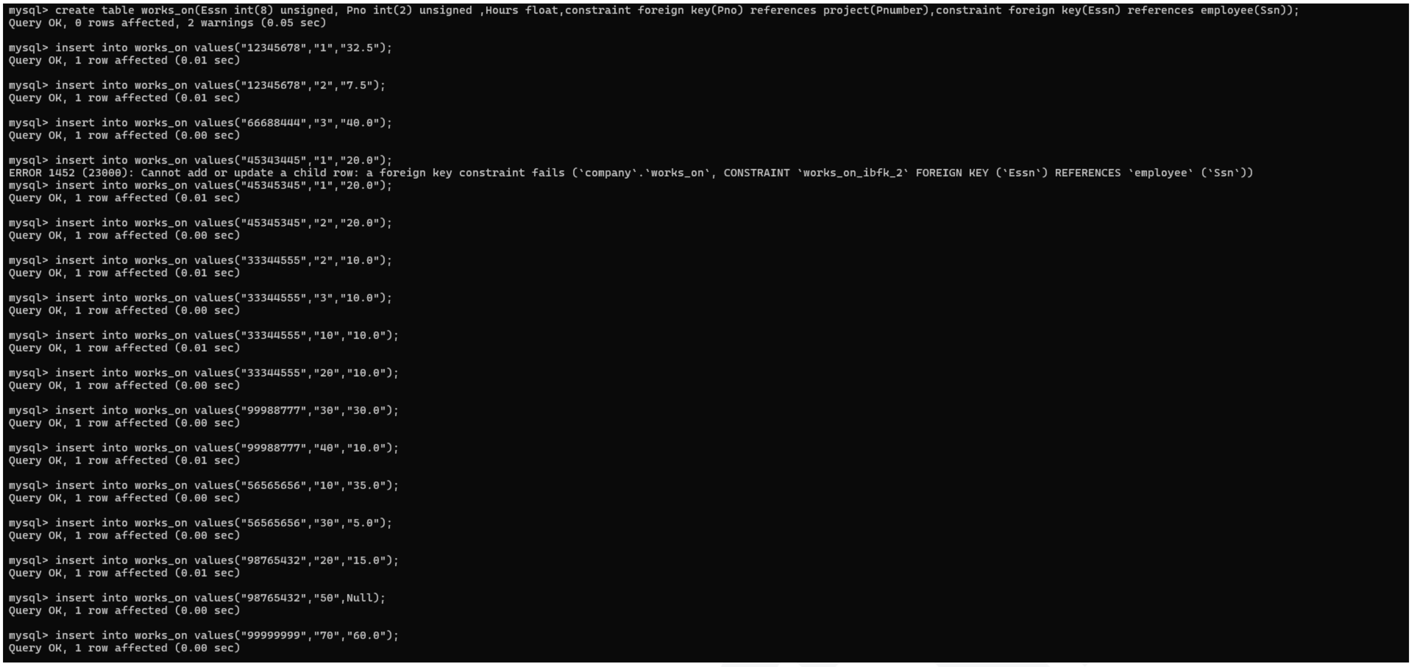
Inserting Values in it:



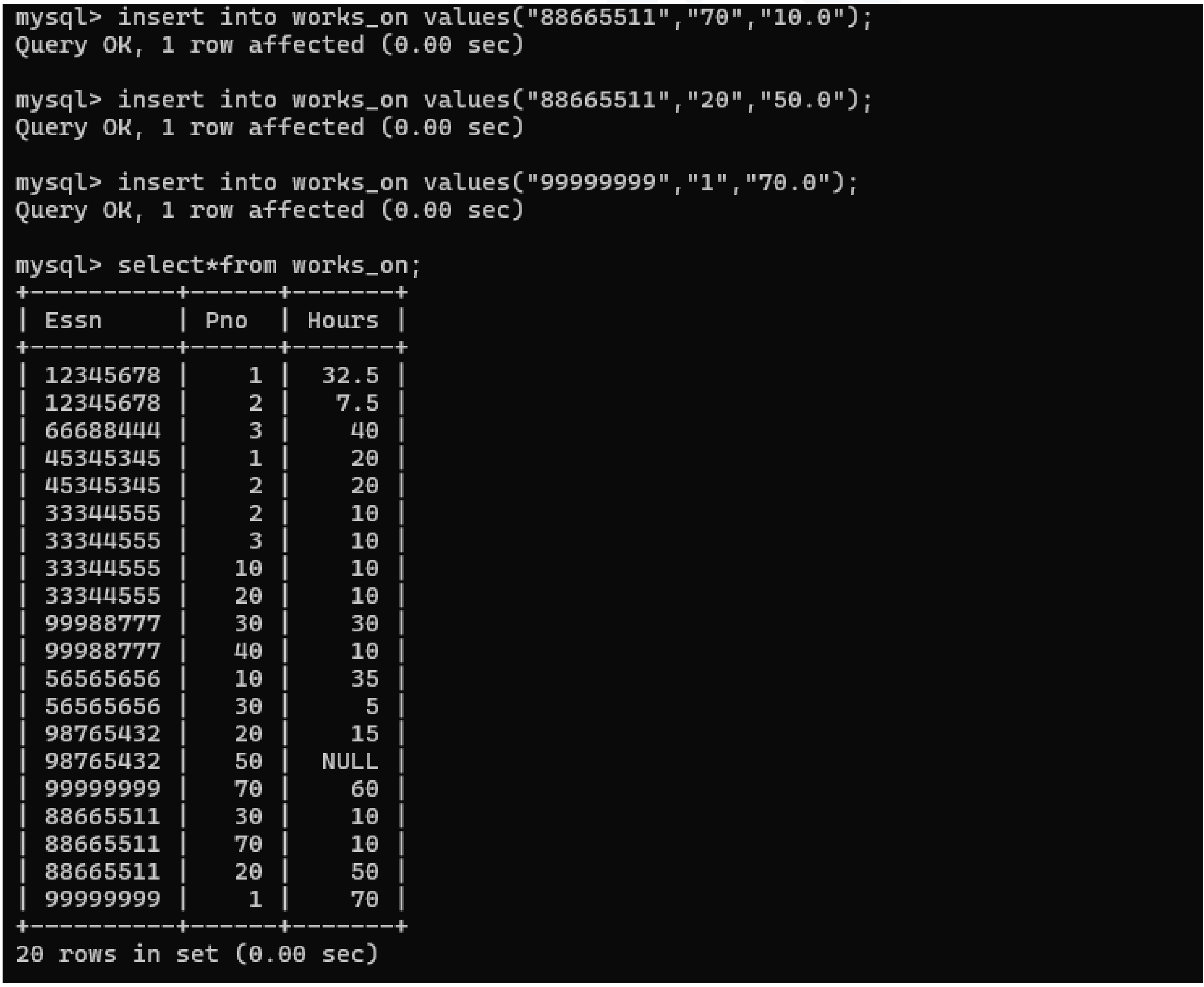
```
mysql> create table dept_location (Dnumber char(3) , Dlocation varchar(20) Primary Key);
Query OK, 0 rows affected (0.02 sec)

mysql> alter table dept_location add constraint foreign key(Dnumber) references department(Dnumber);
Query OK, 0 rows affected (0.08 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> insert into dept_location values("1","Houston");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dept_location values("4","Stafford");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dept_location values("5","Bellaire");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dept_location values("3","Sugarland");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dept_location values("2","NewJercy");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dept_location values("0","Oxford");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dept_location values("0","IJS");
Query OK, 1 row affected (0.01 sec)

mysql> select*from dept_location;
+---------+-----------+
| Dnumber | Dlocation |
+---------+-----------+
| 0       | IJS       |
| 0       | Oxford    |
| 1       | Houston   |
| 2       | NewJercy  |
| 3       | Sugarland |
| 4       | Stafford  |
| 5       | Bellaire  |
+---------+-----------+
7 rows in set (0.00 sec)
```

Creating The Works_On Table.

Inserting Values in it:



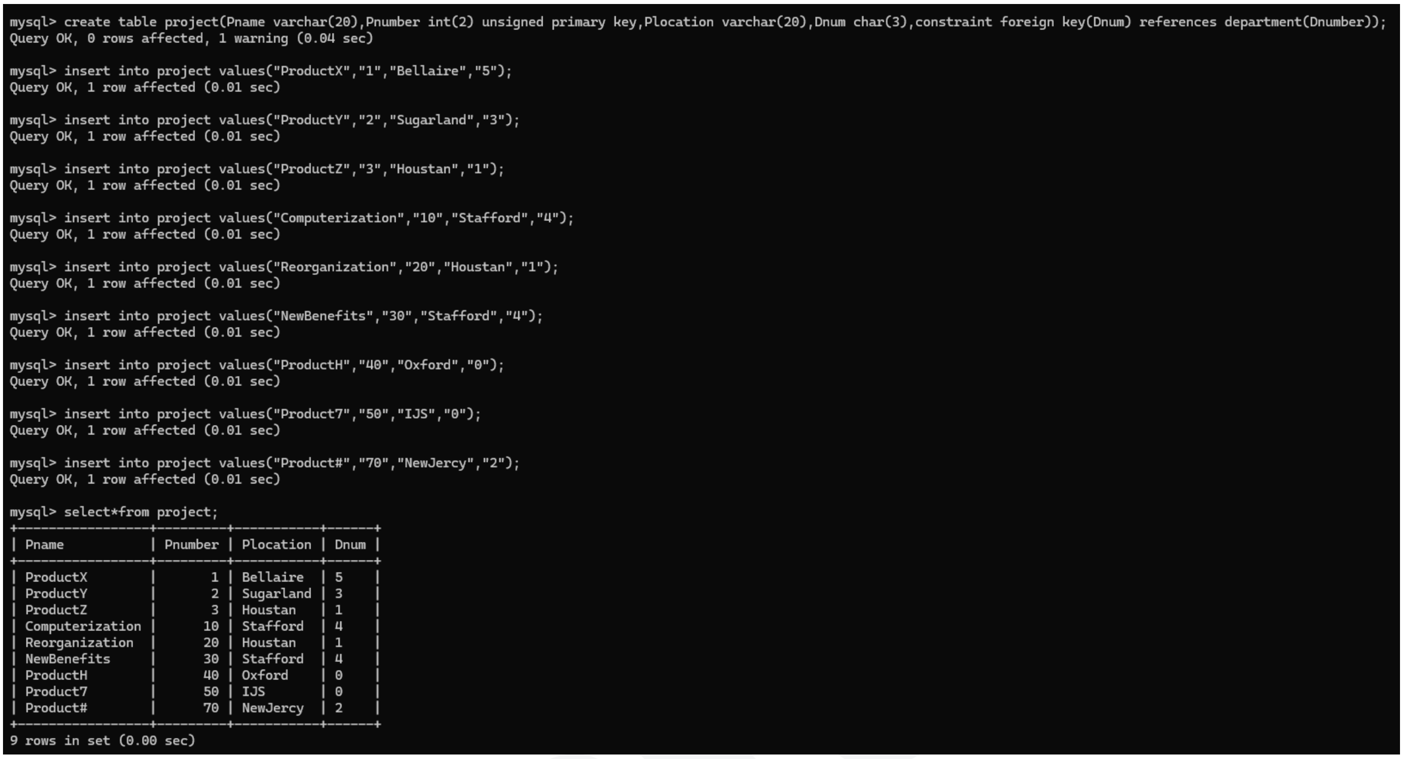
```
mysql> create table works_on(Essn int(8) unsigned, Pno int(2) unsigned ,Hours float,constraint foreign key(Pno) references project(Pnumber),constraint foreign key(Essn) references employee(Ssn));
Query OK, 0 rows affected, 2 warnings (0.05 sec)

mysql> insert into works_on values("12345678","1","32.5");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("12345678","2","7.5");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("66688444","3","40.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("45343445","1","20.0");
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`company`.`works_on`, CONSTRAINT `works_on_ibfk_2` FOREIGN KEY (`Essn`) REFERENCES `employee` (`Ssn`))
mysql> insert into works_on values("45345345","1","20.0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("45345345","2","20.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("33344555","2","10.0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("33344555","3","10.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("33344555","10","10.0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("33344555","20","10.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("99988777","30","30.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("99988777","40","10.0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("56565656","10","35.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("56565656","30","5.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("98765432","20","15.0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into works_on values("98765432","50",Null);
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("99999999","70","60.0");
Query OK, 1 row affected (0.00 sec)
```



```
mysql> insert into works_on values("88665511","70","10.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("88665511","20","50.0");
Query OK, 1 row affected (0.00 sec)

mysql> insert into works_on values("99999999","1","70.0");
Query OK, 1 row affected (0.00 sec)

mysql> select*from works_on;
+----------+------+-------+
| Essn     | Pno  | Hours |
+----------+------+-------+
| 12345678 |    1 |  32.5 |
| 12345678 |    2 |   7.5 |
| 66688444 |    3 |    40 |
| 45345345 |    1 |    20 |
| 45345345 |    2 |    20 |
| 33344555 |    2 |    10 |
| 33344555 |    3 |    10 |
| 33344555 |   10 |    10 |
| 33344555 |   20 |    10 |
| 99988777 |   30 |    30 |
| 99988777 |   40 |    10 |
| 56565656 |   10 |    35 |
| 56565656 |   30 |     5 |
| 98765432 |   20 |    15 |
| 98765432 |   50 |  NULL |
| 99999999 |   70 |    60 |
| 88665511 |   30 |    10 |
| 88665511 |   70 |    10 |
| 88665511 |   20 |    50 |
| 99999999 |    1 |    70 |
+----------+------+-------+
20 rows in set (0.00 sec)
```

Creating The Project Table.

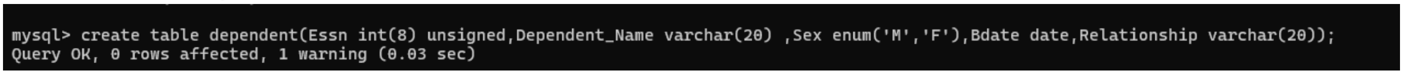
Inserting Values in it:



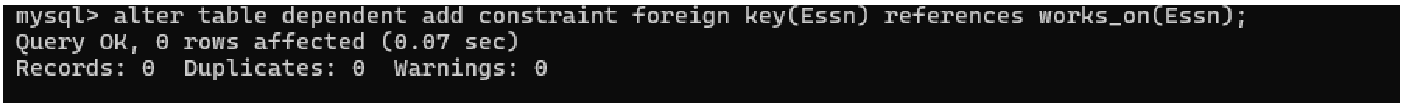
```
mysql> create table project(Pname varchar(20),Pnumber int(2) unsigned primary key,Plocation varchar(20),Dnum char(3),constraint foreign key(Dnum) references department(Dnumber));
Query OK, 0 rows affected, 1 warning (0.04 sec)

mysql> insert into project values("ProductX","1","Bellaire","5");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("ProductY","2","Sugarland","3");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("ProductZ","3","Houstan","1");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("Computerization","10","Stafford","4");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("Reorganization","20","Houstan","1");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("NewBenefits","30","Stafford","4");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("ProductH","40","Oxford","0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("Product7","50","IJS","0");
Query OK, 1 row affected (0.01 sec)

mysql> insert into project values("Product#","70","NewJercy","2");
Query OK, 1 row affected (0.01 sec)

mysql> select*from project;
+-----------------+---------+-----------+------+
| Pname           | Pnumber | Plocation | Dnum |
+-----------------+---------+-----------+------+
| ProductX        |       1 | Bellaire  | 5    |
| ProductY        |       2 | Sugarland | 3    |
| ProductZ        |       3 | Houstan   | 1    |
| Computerization |      10 | Stafford  | 4    |
| Reorganization  |      20 | Houstan   | 1    |
| NewBenefits     |      30 | Stafford  | 4    |
| ProductH        |      40 | Oxford    | 0    |
| Product7        |      50 | IJS       | 0    |
| Product#        |      70 | NewJercy  | 2    |
+-----------------+---------+-----------+------+
9 rows in set (0.00 sec)
```

Creating The Dependent Table.

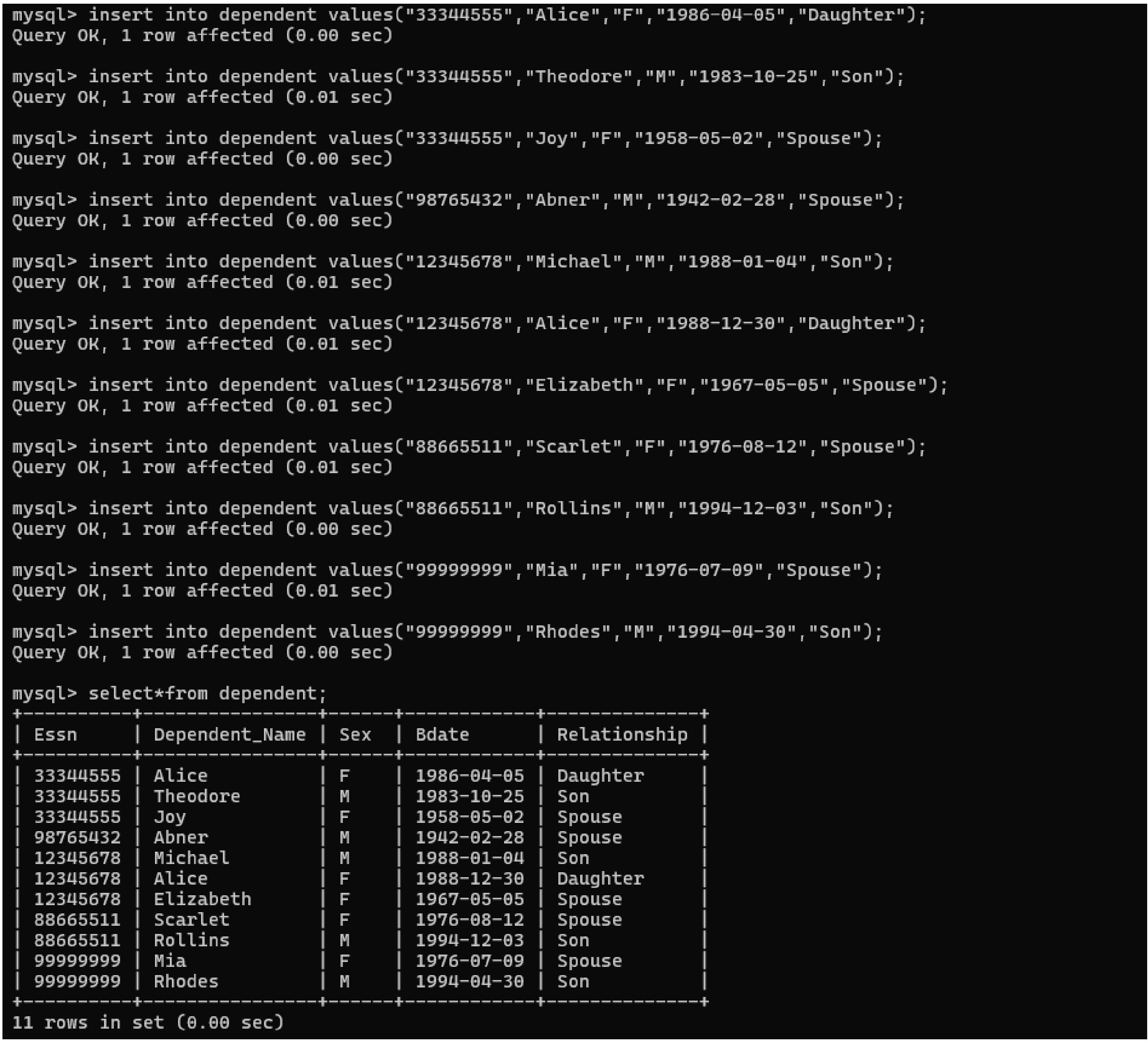
Inserting Values in it:



```
mysql> create table dependent(Essn int(8) unsigned,Dependent_Name varchar(20) ,Sex enum('M','F'),Bdate date,Relationship varchar(20));
Query OK, 0 rows affected, 1 warning (0.03 sec)
```



```
mysql> alter table dependent add constraint foreign key(Essn) references works_on(Essn);
Query OK, 0 rows affected (0.07 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

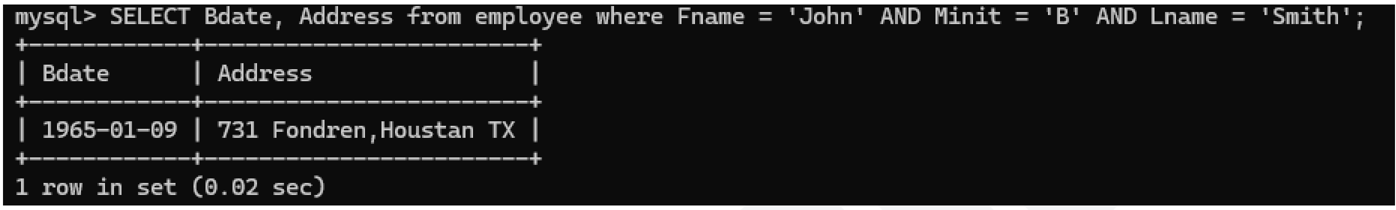


```
mysql> insert into dependent values("33344555","Alice","F","1986-04-05","Daughter");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dependent values("33344555","Theodore","M","1983-10-25","Son");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("33344555","Joy","F","1958-05-02","Spouse");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dependent values("98765432","Abner","M","1942-02-28","Spouse");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dependent values("12345678","Michael","M","1988-01-04","Son");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("12345678","Alice","F","1988-12-30","Daughter");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("12345678","Elizabeth","F","1967-05-05","Spouse");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("88665511","Scarlet","F","1976-08-12","Spouse");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("88665511","Rollins","M","1994-12-03","Son");
Query OK, 1 row affected (0.00 sec)

mysql> insert into dependent values("99999999","Mia","F","1976-07-09","Spouse");
Query OK, 1 row affected (0.01 sec)

mysql> insert into dependent values("99999999","Rhodes","M","1994-04-30","Son");
Query OK, 1 row affected (0.00 sec)

mysql> select*from dependent;
+----------+----------------+------+------------+--------------+
| Essn     | Dependent_Name | Sex  | Bdate      | Relationship |
+----------+----------------+------+------------+--------------+
| 33344555 | Alice          | F    | 1986-04-05 | Daughter     |
| 33344555 | Theodore       | M    | 1983-10-25 | Son          |
| 33344555 | Joy            | F    | 1958-05-02 | Spouse       |
| 98765432 | Abner          | M    | 1942-02-28 | Spouse       |
| 12345678 | Michael        | M    | 1988-01-04 | Son          |
| 12345678 | Alice          | F    | 1988-12-30 | Daughter     |
| 12345678 | Elizabeth      | F    | 1967-05-05 | Spouse       |
| 88665511 | Scarlet        | F    | 1976-08-12 | Spouse       |
| 88665511 | Rollins        | M    | 1994-12-03 | Son          |
| 99999999 | Mia            | F    | 1976-07-09 | Spouse       |
| 99999999 | Rhodes         | M    | 1994-04-30 | Son          |
+----------+----------------+------+------------+--------------+
11 rows in set (0.00 sec)
```

Queries to do:

Query 0. Retrieve the birth date and address of the employee(s) whose name is ‘John B. Smith’.

Output:



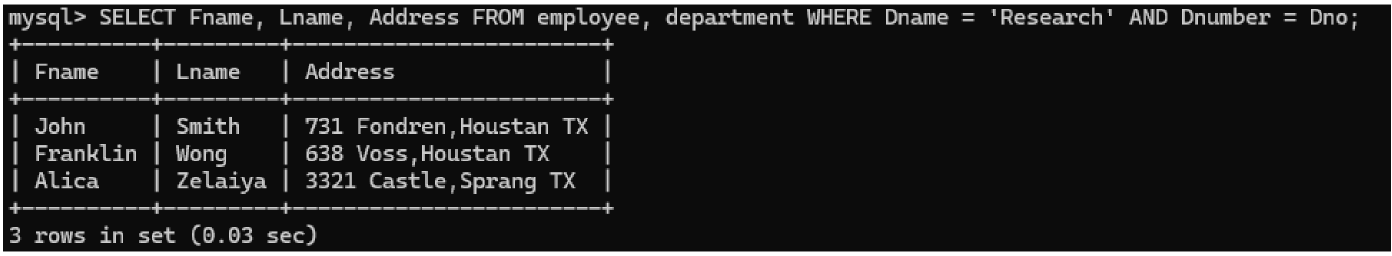
```
mysql> SELECT Bdate, Address from employee where Fname = 'John' AND Minit = 'B' AND Lname = 'Smith';
+------------+-------------------------+
| Bdate      | Address                 |
+------------+-------------------------+
| 1965-01-09 | 731 Fondren,Houstan TX |
+------------+-------------------------+
1 row in set (0.02 sec)
```

Query 1. Retrieve the name and address

of all employees who work for the

‘Research’ department.

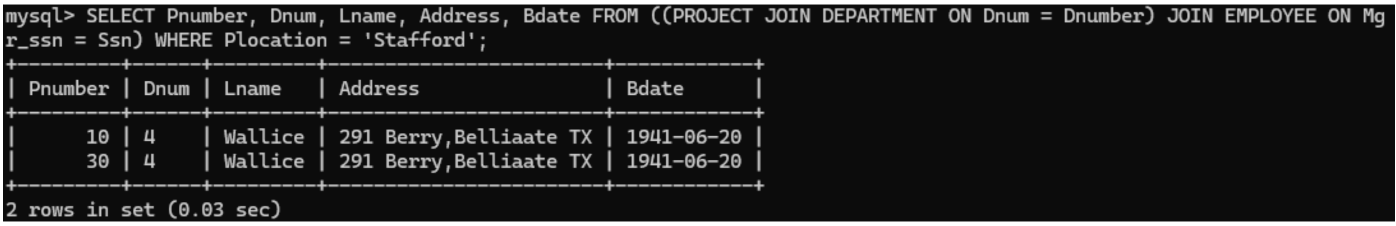
Output:



```
mysql> SELECT Fname, Lname, Address FROM employee, department WHERE Dname = 'Research' AND Dnumber = Dno;
+----------+---------+-------------------------+
| Fname    | Lname   | Address                 |
+----------+---------+-------------------------+
| John     | Smith   | 731 Fondren,Houstan TX |
| Franklin | Wong    | 638 Voss,Houstan TX    |
| Alica    | Zelaiya | 3321 Castle,Sprang TX  |
+----------+---------+-------------------------+
3 rows in set (0.03 sec)
```

Query 2. For every project located in ‘Stafford’, list the project number, the controlling department number, and the department manager’s last name, address, and birth date.

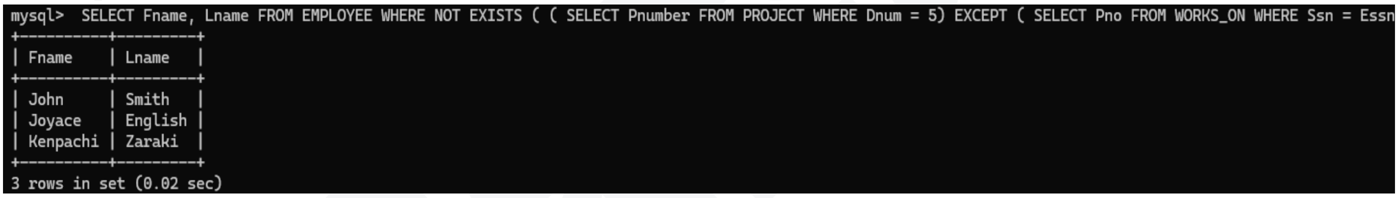
Output:



```
mysql> SELECT Pnumber, Dnum, Lname, Address, Bdate FROM ((PROJECT JOIN DEPARTMENT ON Dnum = Dnumber) JOIN EMPLOYEE ON Mg
r_ssn = Ssn) WHERE Plocation = 'Stafford';
+---------+------+---------+------------------------+------------+
| Pnumber | Dnum | Lname   | Address                | Bdate      |
+---------+------+---------+------------------------+------------+
|      10 | 4    | Wallice | 291 Berry,Belliaate TX | 1941-06-20 |
|      30 | 4    | Wallice | 291 Berry,Belliaate TX | 1941-06-20 |
+---------+------+---------+------------------------+------------+
2 rows in set (0.03 sec)
```

Query Q3: Retrieve the name of each employee who works on all the projects controlled by department number 5

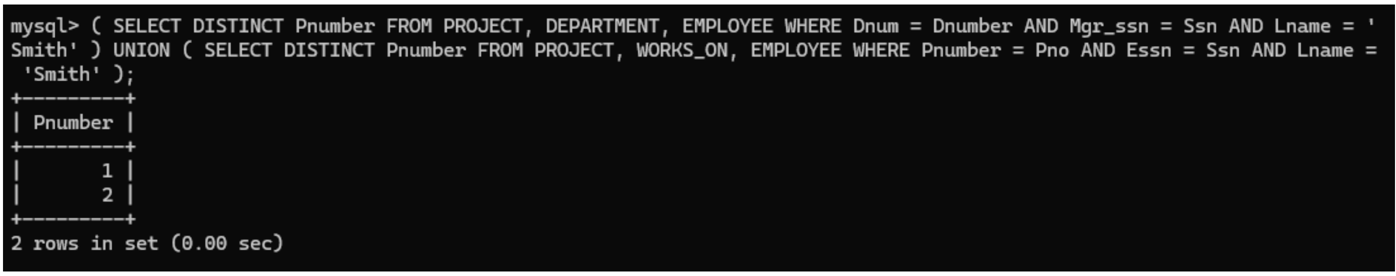
Output:



```
mysql>  SELECT Fname, Lname FROM EMPLOYEE WHERE NOT EXISTS ( ( SELECT Pnumber FROM PROJECT WHERE Dnum = 5) EXCEPT ( SELECT Pno FROM WORKS_ON WHERE Ssn = Essn
+----------+---------+
| Fname    | Lname   |
+----------+---------+
| John     | Smith   |
| Joyace   | English |
| Kenpachi | Zaraki  |
+----------+---------+
3 rows in set (0.02 sec)
```

Query 4:Make a list of all project numbers for projects that involve an employee whose last name is ‘Smith’, either as a worker or as a manager of the department that controls the project

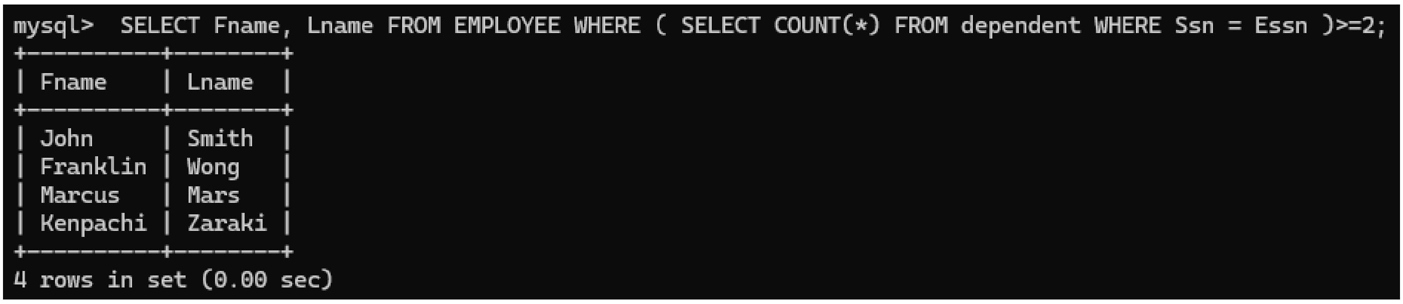
Output:



```
mysql> ( SELECT DISTINCT Pnumber FROM PROJECT, DEPARTMENT, EMPLOYEE WHERE Dnum = Dnumber AND Mgr_ssn = Ssn AND Lname = '
Smith' ) UNION ( SELECT DISTINCT Pnumber FROM PROJECT, WORKS_ON, EMPLOYEE WHERE Pnumber = Pno AND Essn = Ssn AND Lname =
 'Smith' );
+---------+
| Pnumber |
+---------+
|       1 |
|       2 |
+---------+
2 rows in set (0.00 sec)
```

Query 5:To retrieve the names of all employees who have two or more dependents

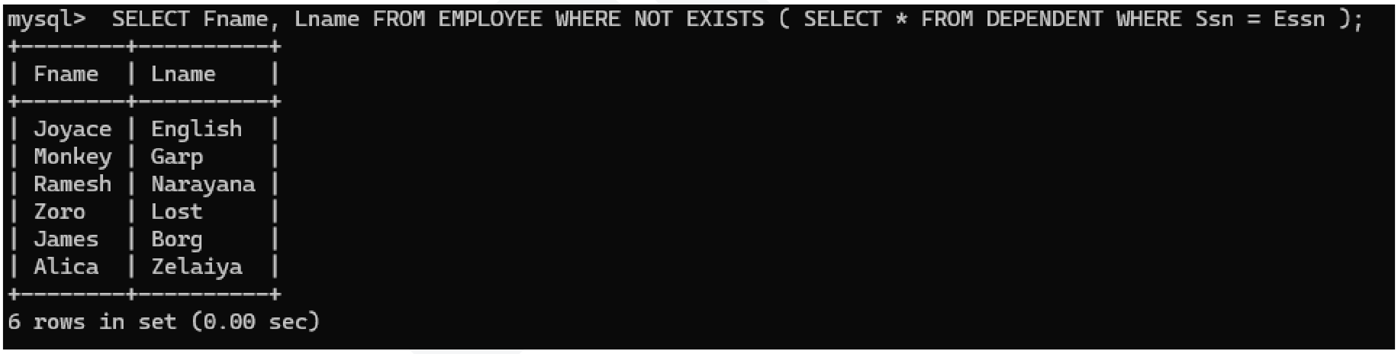
Output:



```
mysql>  SELECT Fname, Lname FROM EMPLOYEE WHERE ( SELECT COUNT(*) FROM dependent WHERE Ssn = Essn )>=2;
+----------+--------+
| Fname    | Lname  |
+----------+--------+
| John     | Smith  |
| Franklin | Wong   |
| Marcus   | Mars   |
| Kenpachi | Zaraki |
+----------+--------+
4 rows in set (0.00 sec)
```

Query 6: Retrieve the names of employees who have no dependents.

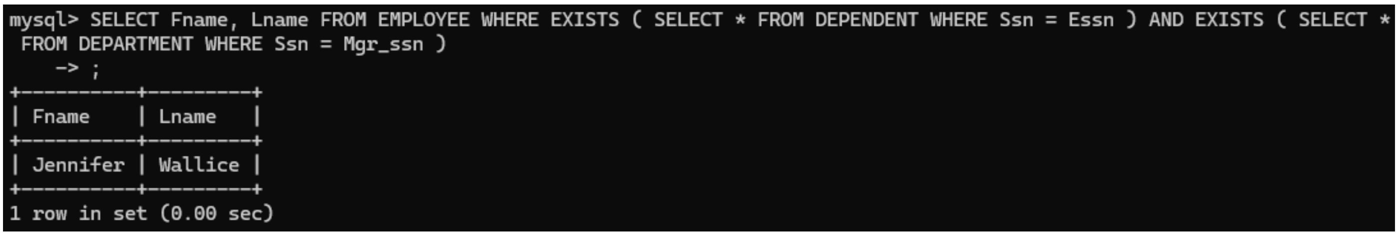
Output:



```
mysql>  SELECT Fname, Lname FROM EMPLOYEE WHERE NOT EXISTS ( SELECT * FROM DEPENDENT WHERE Ssn = Essn );
+--------+----------+
| Fname  | Lname    |
+--------+----------+
| Joyace | English  |
| Monkey | Garp     |
| Ramesh | Narayana |
| Zoro   | Lost     |
| James  | Borg     |
| Alica  | Zelaiya  |
+--------+----------+
6 rows in set (0.00 sec)
```

Query 7. List the names of managers who have at least one dependent.

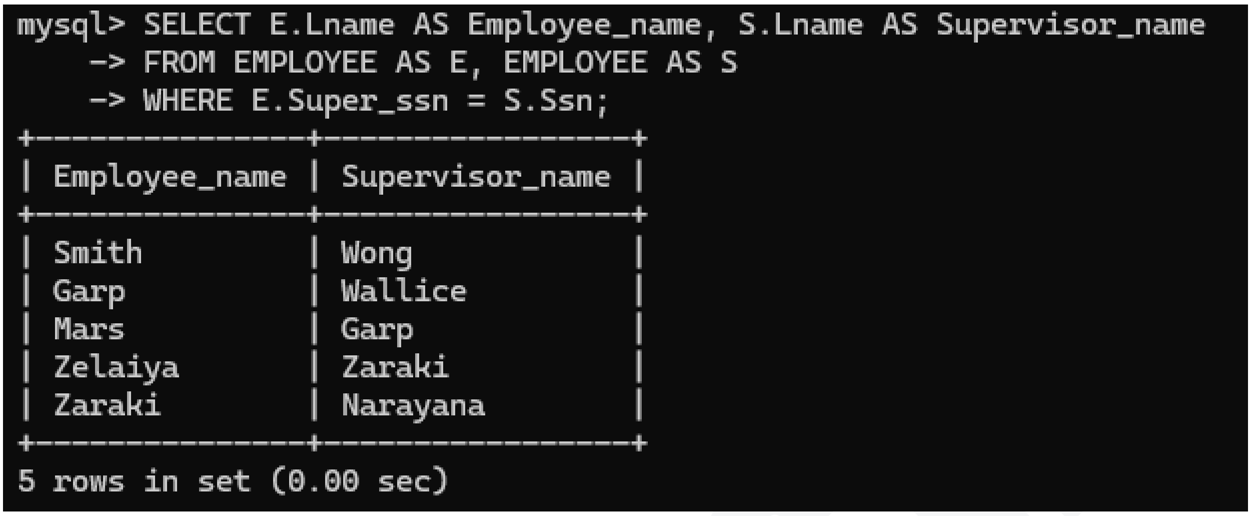
Output:



```
mysql> SELECT Fname, Lname FROM EMPLOYEE WHERE EXISTS ( SELECT * FROM DEPENDENT WHERE Ssn = Essn ) AND EXISTS ( SELECT *
 FROM DEPARTMENT WHERE Ssn = Mgr_ssn )
    -> ;
+----------+---------+
| Fname    | Lname   |
+----------+---------+
| Jennifer | Wallice |
+----------+---------+
1 row in set (0.00 sec)
```

Query 8. For each employee, retrieve the employee's first and last name and the first and last name of his or her immediate supervisor.

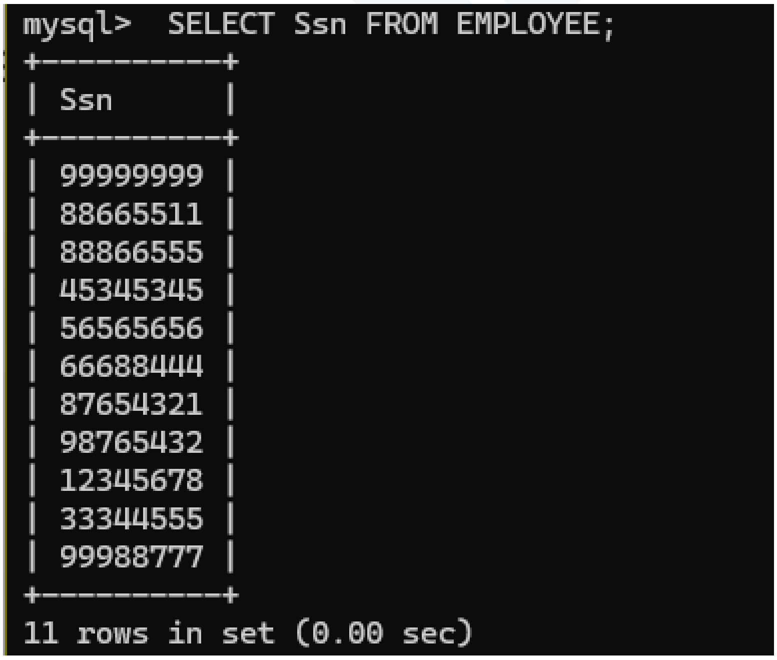
Output:



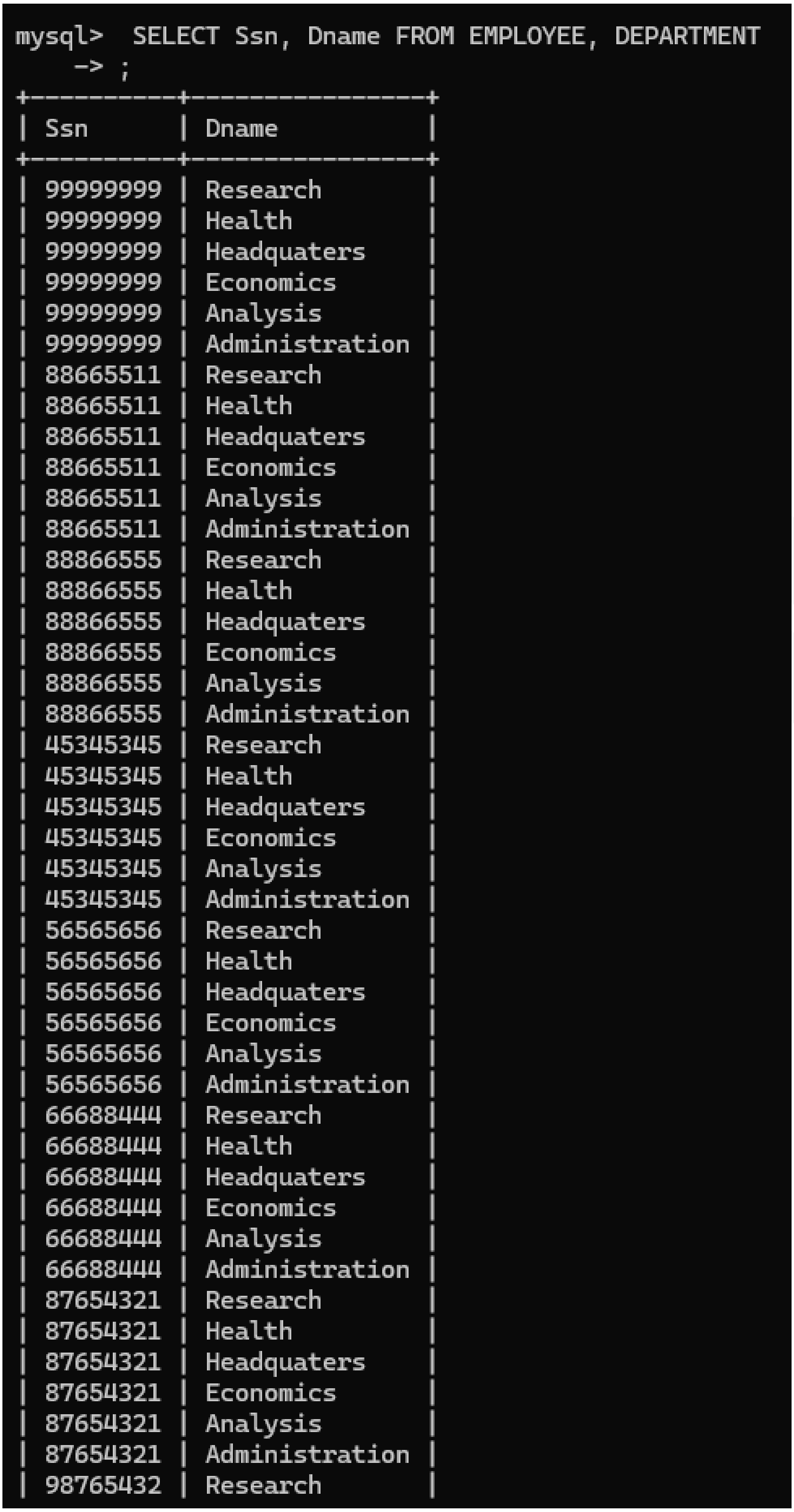
```
mysql> SELECT E.Lname AS Employee_name, S.Lname AS Supervisor_name
    -> FROM EMPLOYEE AS E, EMPLOYEE AS S
    -> WHERE E.Super_ssn = S.Ssn;
+---------------+-----------------+
| Employee_name | Supervisor_name |
+---------------+-----------------+
| Smith         | Wong            |
| Garp          | Wallice         |
| Mars          | Garp            |
| Zelaiya       | Zaraki          |
| Zaraki        | Narayana        |
+---------------+-----------------+
5 rows in set (0.00 sec)
```

Queries 9 and 10. Select all EMPLOYEE Ssns (Q9) and all combinations of EMPLOYEE Ssn and DEPARTMENT Dname (Q10) in the database.

Output:9,10



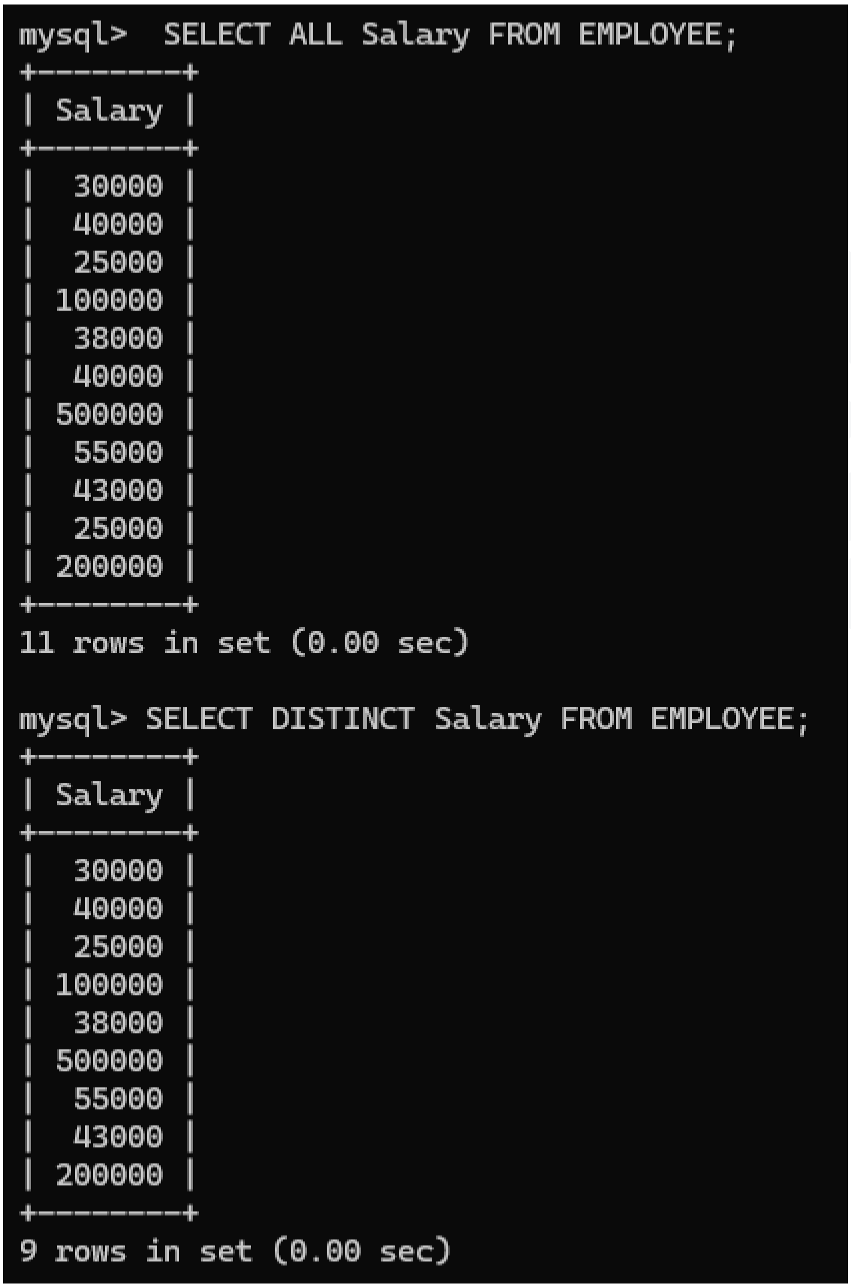
```
mysql>  SELECT Ssn FROM EMPLOYEE;
+----------+
| Ssn      |
+----------+
| 99999999 |
| 88665511 |
| 88866555 |
| 45345345 |
| 56565656 |
| 66688444 |
| 87654321 |
| 98765432 |
| 12345678 |
| 33344555 |
| 99988777 |
+----------+
11 rows in set (0.00 sec)
```



```
mysql>  SELECT Ssn, Dname FROM EMPLOYEE, DEPARTMENT
    -> ;
+----------+----------------+
| Ssn      | Dname          |
+----------+----------------+
| 99999999 | Research       |
| 99999999 | Health         |
| 99999999 | Headquaters    |
| 99999999 | Economics      |
| 99999999 | Analysis       |
| 99999999 | Administration |
| 88665511 | Research       |
| 88665511 | Health         |
| 88665511 | Headquaters    |
| 88665511 | Economics      |
| 88665511 | Analysis       |
| 88665511 | Administration |
| 88866555 | Research       |
| 88866555 | Health         |
| 88866555 | Headquaters    |
| 88866555 | Economics      |
| 88866555 | Analysis       |
| 88866555 | Administration |
| 45345345 | Research       |
| 45345345 | Health         |
| 45345345 | Headquaters    |
| 45345345 | Economics      |
| 45345345 | Analysis       |
| 45345345 | Administration |
| 56565656 | Research       |
| 56565656 | Health         |
| 56565656 | Headquaters    |
| 56565656 | Economics      |
| 56565656 | Analysis       |
| 56565656 | Administration |
| 66688444 | Research       |
| 66688444 | Health         |
| 66688444 | Headquaters    |
| 66688444 | Economics      |
| 66688444 | Analysis       |
| 66688444 | Administration |
| 87654321 | Research       |
| 87654321 | Health         |
| 87654321 | Headquaters    |
| 87654321 | Economics      |
| 87654321 | Analysis       |
| 87654321 | Administration |
| 98765432 | Research       |
```

Query 11. Retrieve the salary of every employee (Q11) and all distinct salary values (Q11A).

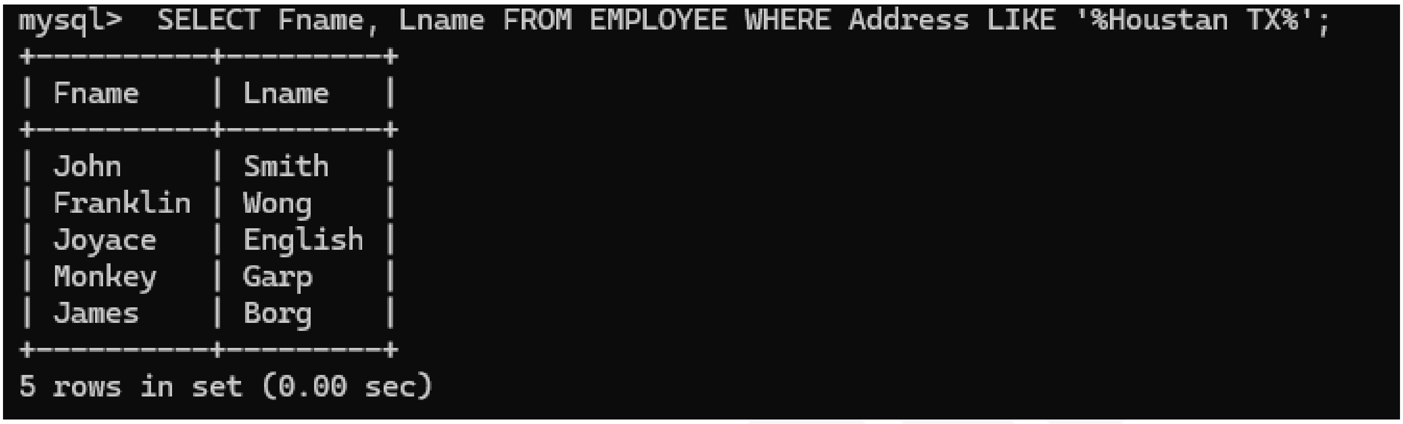
Output:



```
mysql>  SELECT ALL Salary FROM EMPLOYEE;
+--------+
| Salary |
+--------+
|  30000 |
|  40000 |
|  25000 |
| 100000 |
|  38000 |
|  40000 |
| 500000 |
|  55000 |
|  43000 |
|  25000 |
| 200000 |
+--------+
11 rows in set (0.00 sec)

mysql> SELECT DISTINCT Salary FROM EMPLOYEE;
+--------+
| Salary |
+--------+
|  30000 |
|  40000 |
|  25000 |
| 100000 |
|  38000 |
| 500000 |
|  55000 |
|  43000 |
| 200000 |
+--------+
9 rows in set (0.00 sec)
```

Query 12: Retrieve all employees whose address is in Houston, Texas.

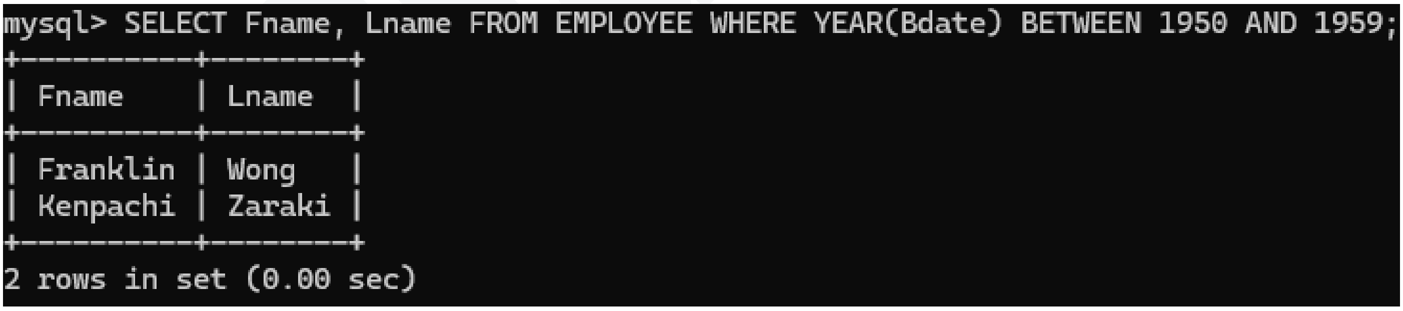
Output:



```
mysql>  SELECT Fname, Lname FROM EMPLOYEE WHERE Address LIKE '%Houstan TX%';
+----------+---------+
| Fname    | Lname   |
+----------+---------+
| John     | Smith   |
| Franklin | Wong    |
| Joyace   | English |
| Monkey   | Garp    |
| James    | Borg    |
+----------+---------+
5 rows in set (0.00 sec)
```

Query 12a). Find all employees who were born during the 1950s. .

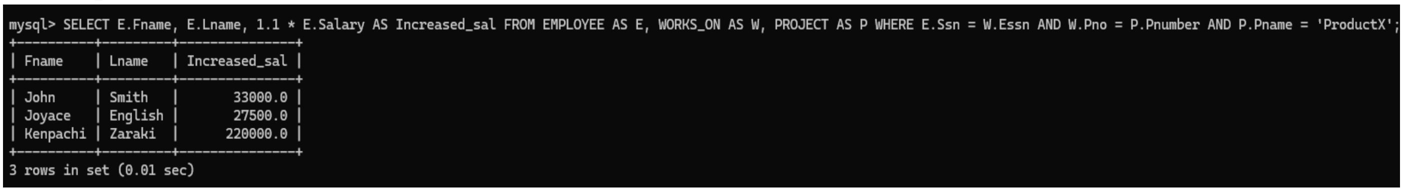
Output:



```
mysql> SELECT Fname, Lname FROM EMPLOYEE WHERE YEAR(Bdate) BETWEEN 1950 AND 1959;
+----------+--------+
| Fname    | Lname  |
+----------+--------+
| Franklin | Wong   |
| Kenpachi | Zaraki |
+----------+--------+
2 rows in set (0.00 sec)
```

Query 13. Show the resulting salaries if every employee working on the 'ProductX' project is given a 10% raise

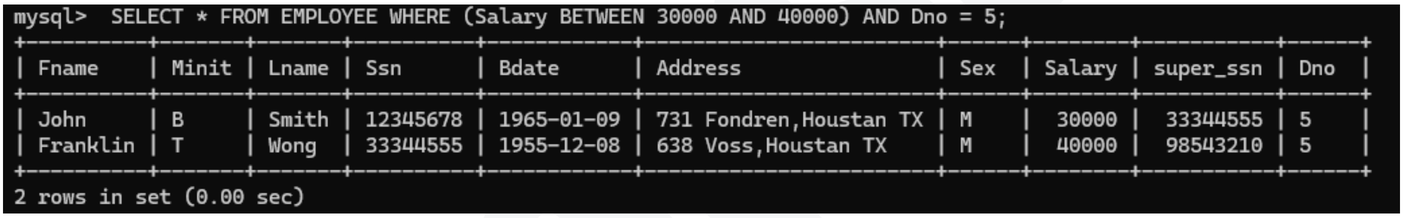
Output:



```
mysql> SELECT E.Fname, E.Lname, 1.1 * E.Salary AS Increased_sal FROM EMPLOYEE AS E, WORKS_ON AS W, PROJECT AS P WHERE E.Ssn = W.Essn AND W.Pno = P.Pnumber AND P.Pname = 'ProductX';
+----------+---------+---------------+
| Fname    | Lname   | Increased_sal |
+----------+---------+---------------+
| John     | Smith   |       33000.0 |
| Joyace   | English |       27500.0 |
| Kenpachi | Zaraki  |      220000.0 |
+----------+---------+---------------+
3 rows in set (0.01 sec)
```

Query 14. Retrieve all employees in department 5 whose salary is between $30,000 and $40,000.

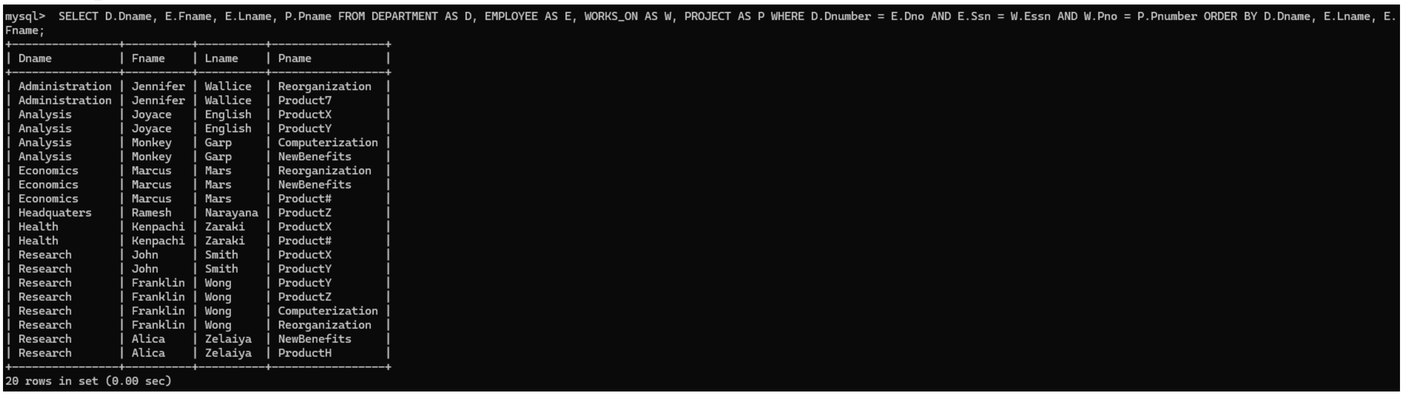
Output:



```
mysql>  SELECT * FROM EMPLOYEE WHERE (Salary BETWEEN 30000 AND 40000) AND Dno = 5;
+----------+-------+-------+----------+------------+------------------------+------+--------+-----------+------+
| Fname    | Minit | Lname | Ssn      | Bdate      | Address                | Sex  | Salary | super_ssn | Dno  |
+----------+-------+-------+----------+------------+------------------------+------+--------+-----------+------+
| John     | B     | Smith | 12345678 | 1965-01-09 | 731 Fondren,Houstan TX | M    |  30000 |  33344555 | 5    |
| Franklin | T     | Wong  | 33344555 | 1955-12-08 | 638 Voss,Houstan TX    | M    |  40000 |  98543210 | 5    |
+----------+-------+-------+----------+------------+------------------------+------+--------+-----------+------+
2 rows in set (0.00 sec)
```

Query 15:Retrieve a list of employees and the projects they are working on, ordered by department and, within each department, ordered alphabetically by last name, then first name.

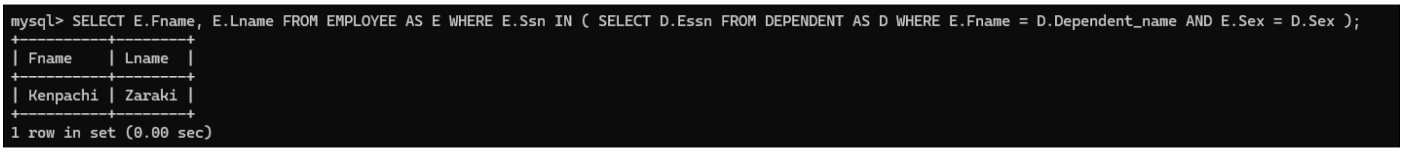
Output:



```
mysql>  SELECT D.Dname, E.Fname, E.Lname, P.Pname FROM DEPARTMENT AS D, EMPLOYEE AS E, WORKS_ON AS W, PROJECT AS P WHERE D.Dnumber = E.Dno AND E.Ssn = W.Essn AND W.Pno = P.Pnumber ORDER BY D.Dname, E.Lname, E.
Fname;
+----------------+----------+----------+-----------------+
| Dname          | Fname    | Lname    | Pname           |
+----------------+----------+----------+-----------------+
| Administration | Jennifer | Wallice  | Reorganization  |
| Administration | Jennifer | Wallice  | Product7        |
| Analysis       | Joyace   | English  | ProductX        |
| Analysis       | Joyace   | English  | ProductY        |
| Analysis       | Monkey   | Garp     | Computerization |
| Analysis       | Monkey   | Garp     | NewBenefits     |
| Economics      | Marcus   | Mars     | Reorganization  |
| Economics      | Marcus   | Mars     | NewBenefits     |
| Economics      | Marcus   | Mars     | Product#        |
| Headquaters    | Ramesh   | Narayana | ProductZ        |
| Health         | Kenpachi | Zaraki   | ProductX        |
| Health         | Kenpachi | Zaraki   | Product#        |
| Research       | John     | Smith    | ProductX        |
| Research       | John     | Smith    | ProductY        |
| Research       | Franklin | Wong     | ProductY        |
| Research       | Franklin | Wong     | ProductZ        |
| Research       | Franklin | Wong     | Computerization |
| Research       | Franklin | Wong     | Reorganization  |
| Research       | Alica    | Zelaiya  | NewBenefits     |
| Research       | Alica    | Zelaiya  | ProductH        |
+----------------+----------+----------+-----------------+
20 rows in set (0.00 sec)
```

Query 16. Retrieve the name of each employee who has a dependent with the same first name and is the same sex as the employee.

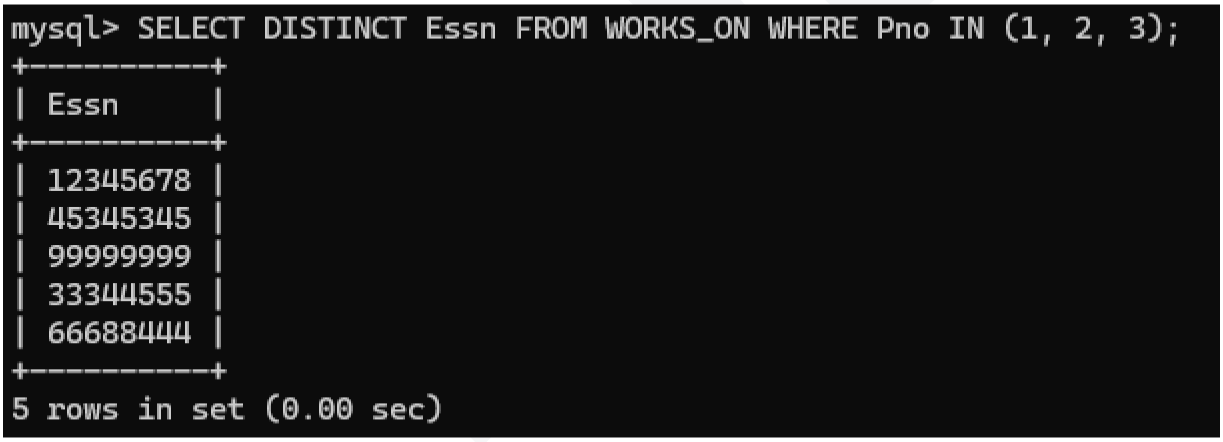
Output:



```
mysql> SELECT E.Fname, E.Lname FROM EMPLOYEE AS E WHERE E.Ssn IN ( SELECT D.Essn FROM DEPENDENT AS D WHERE E.Fname = D.Dependent_name AND E.Sex = D.Sex );
+----------+--------+
| Fname    | Lname  |
+----------+--------+
| Kenpachi | Zaraki |
+----------+--------+
1 row in set (0.00 sec)
```

Query 17. Retrieve the Social Security numbers of all employees who work on project numbers 1, 2, or 3.

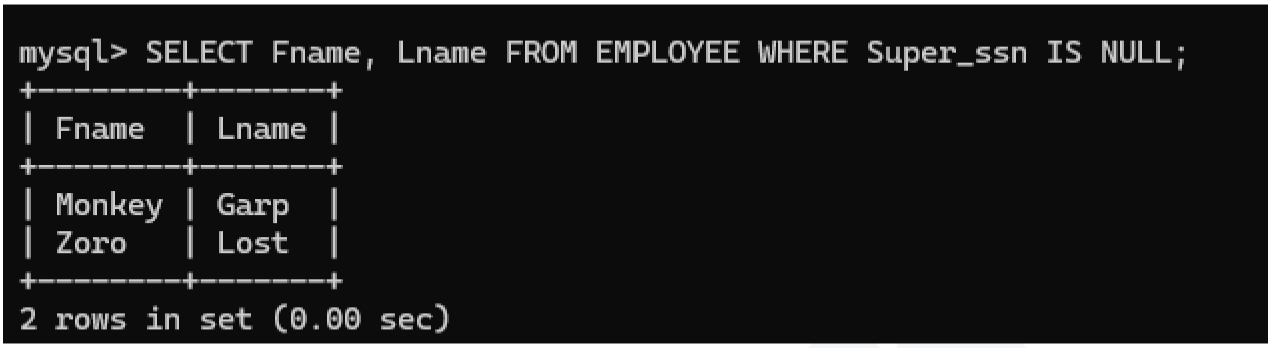
Output:



```
mysql> SELECT DISTINCT Essn FROM WORKS_ON WHERE Pno IN (1, 2, 3);
+----------+
| Essn     |
+----------+
| 12345678 |
| 45345345 |
| 99999999 |
| 33344555 |
| 66688444 |
+----------+
5 rows in set (0.00 sec)
```

Query 18. Retrieve the names of all employees who do not have supervisors.

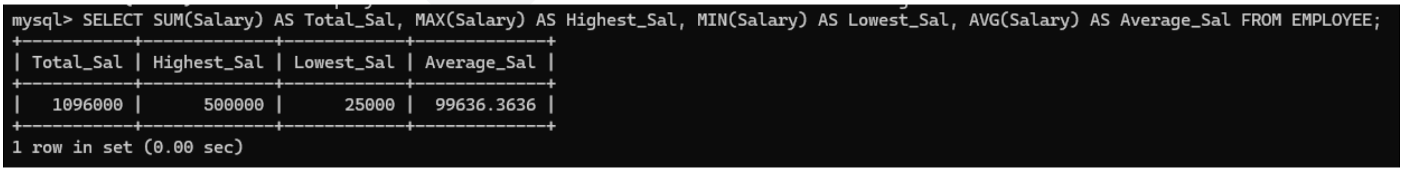
Output:



```
mysql> SELECT Fname, Lname FROM EMPLOYEE WHERE Super_ssn IS NULL;
+--------+-------+
| Fname  | Lname |
+--------+-------+
| Monkey | Garp  |
| Zoro   | Lost  |
+--------+-------+
2 rows in set (0.00 sec)
```

Query 19. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Q19: SELECT SUM (Salary), MAX (Salary), MIN (Salary), AVG (Salary) FROM EMPLOYEE;

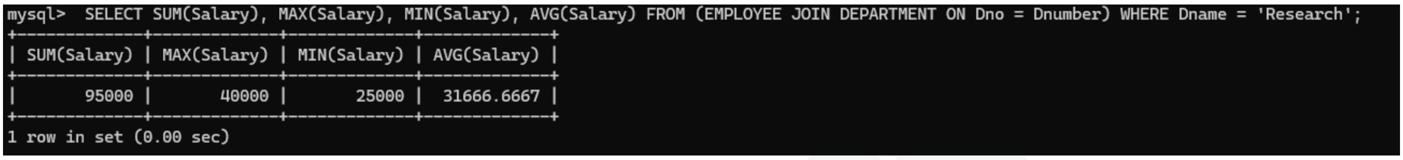
Output:



```
mysql> SELECT SUM(Salary) AS Total_Sal, MAX(Salary) AS Highest_Sal, MIN(Salary) AS Lowest_Sal, AVG(Salary) AS Average_Sal FROM EMPLOYEE;
+-----------+-------------+------------+-------------+
| Total_Sal | Highest_Sal | Lowest_Sal | Average_Sal |
+-----------+-------------+------------+-------------+
|   1096000 |      500000 |      25000 |  99636.3636 |
+-----------+-------------+------------+-------------+
1 row in set (0.00 sec)
```

Query 20. Find the sum of the salaries of all employees of the 'Research' department, as well as the maximum salary, the minimum salary, and the average salary in this department.

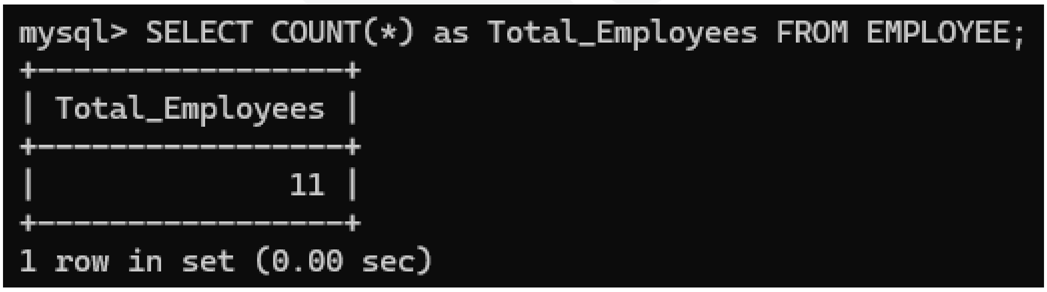
Output:



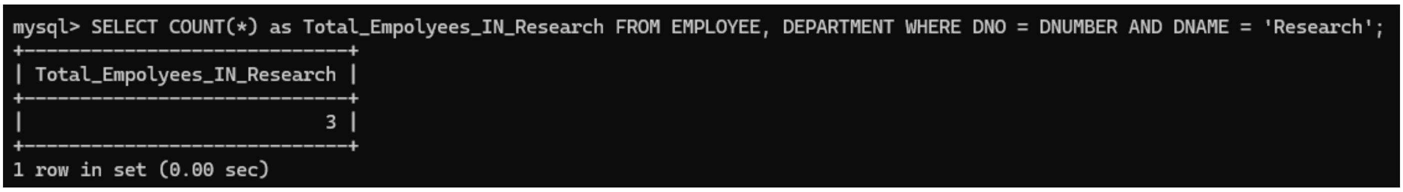
```
mysql>  SELECT SUM(Salary), MAX(Salary), MIN(Salary), AVG(Salary) FROM (EMPLOYEE JOIN DEPARTMENT ON Dno = Dnumber) WHERE Dname = 'Research';
+-------------+-------------+-------------+-------------+
| SUM(Salary) | MAX(Salary) | MIN(Salary) | AVG(Salary) |
+-------------+-------------+-------------+-------------+
|       95000 |       40000 |       25000 |  31666.6667 |
+-------------+-------------+-------------+-------------+
1 row in set (0.00 sec)
```

Queries 21 and 22. Retrieve the total number of employees in the company (Q21) and the number of employees in the 'Research' department (Q22).

Output 21,22:



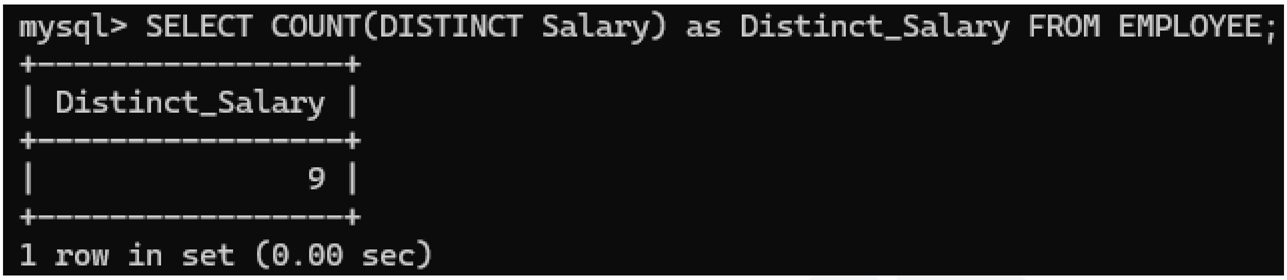
```
mysql> SELECT COUNT(*) as Total_Employees FROM EMPLOYEE;
+-----------------+
| Total_Employees |
+-----------------+
|              11 |
+-----------------+
1 row in set (0.00 sec)
```



```
mysql> SELECT COUNT(*) as Total_Empolyees_IN_Research FROM EMPLOYEE, DEPARTMENT WHERE DNO = DNUMBER AND DNAME = 'Research';
+-----------------------------+
| Total_Empolyees_IN_Research |
+-----------------------------+
|                           3 |
+-----------------------------+
1 row in set (0.00 sec)
```

Query 23. Count the number of distinct salary values in the database.

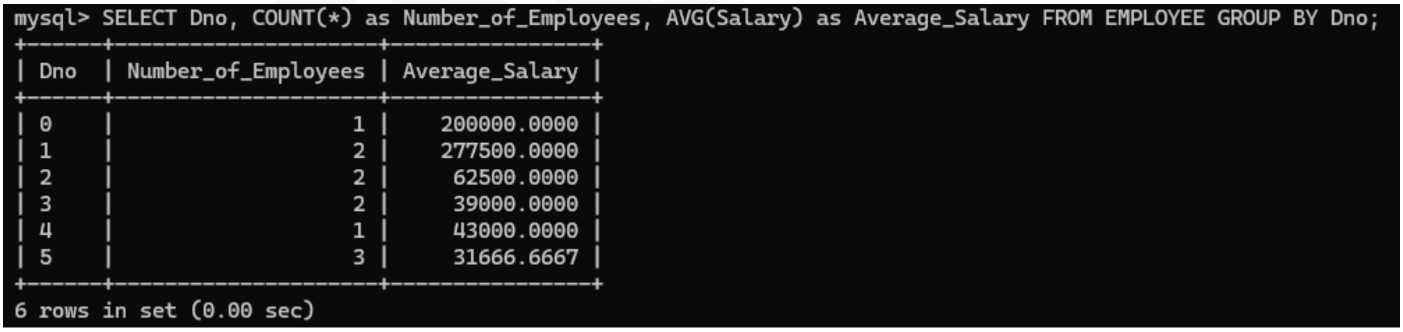
Output:



```
mysql> SELECT COUNT(DISTINCT Salary) as Distinct_Salary FROM EMPLOYEE;
+-----------------+
| Distinct_Salary |
+-----------------+
|               9 |
+-----------------+
1 row in set (0.00 sec)
```

Query 24. For each department, retrieve the department number, the number of employees in the department, and their average salary

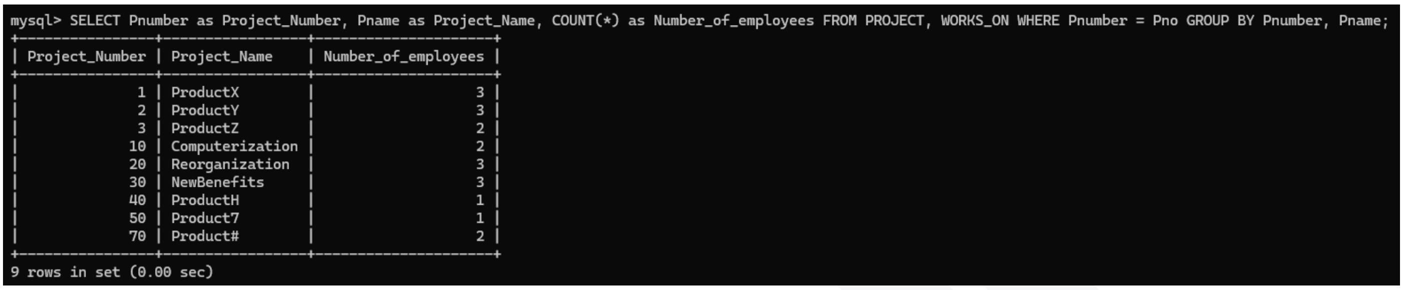
Output:



```
mysql> SELECT Dno, COUNT(*) as Number_of_Employees, AVG(Salary) as Average_Salary FROM EMPLOYEE GROUP BY Dno;
+------+---------------------+----------------+
| Dno  | Number_of_Employees | Average_Salary |
+------+---------------------+----------------+
| 0    |                   1 |    200000.0000 |
| 1    |                   2 |    277500.0000 |
| 2    |                   2 |     62500.0000 |
| 3    |                   2 |     39000.0000 |
| 4    |                   1 |     43000.0000 |
| 5    |                   3 |     31666.6667 |
+------+---------------------+----------------+
6 rows in set (0.00 sec)
```

Query 25. For each project, retrieve the project number, the project name, and the number of employees who work on that project.

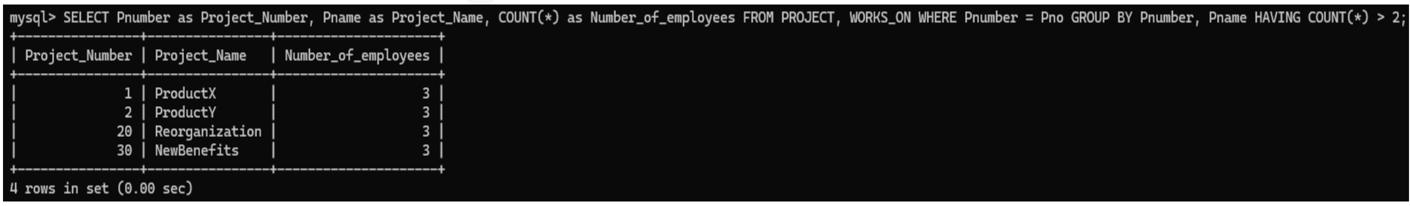
Output:



```
mysql> SELECT Pnumber as Project_Number, Pname as Project_Name, COUNT(*) as Number_of_employees FROM PROJECT, WORKS_ON WHERE Pnumber = Pno GROUP BY Pnumber, Pname;
+----------------+-----------------+---------------------+
| Project_Number | Project_Name    | Number_of_employees |
+----------------+-----------------+---------------------+
|              1 | ProductX        |                   3 |
|              2 | ProductY        |                   3 |
|              3 | ProductZ        |                   2 |
|             10 | Computerization |                   2 |
|             20 | Reorganization  |                   3 |
|             30 | NewBenefits     |                   3 |
|             40 | ProductH        |                   1 |
|             50 | Product7        |                   1 |
|             70 | Product#        |                   2 |
+----------------+-----------------+---------------------+
9 rows in set (0.00 sec)
```

Query 26. For each project on which more than two employees work, retrieve the project number, the project name, and the number of employees who work on the project.

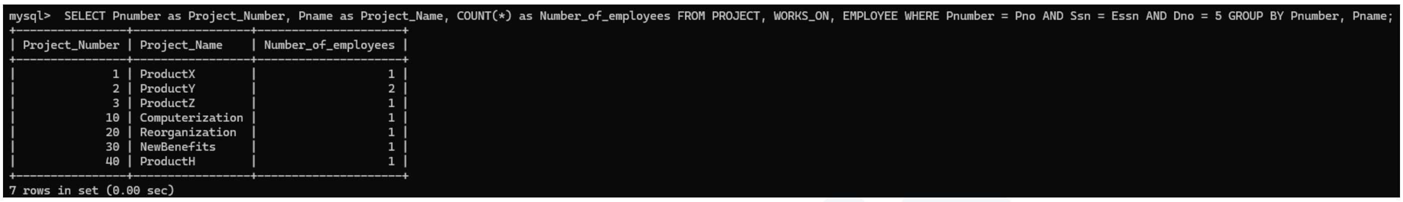
Output:



```
mysql> SELECT Pnumber as Project_Number, Pname as Project_Name, COUNT(*) as Number_of_employees FROM PROJECT, WORKS_ON WHERE Pnumber = Pno GROUP BY Pnumber, Pname HAVING COUNT(*) > 2;
+----------------+----------------+---------------------+
| Project_Number | Project_Name   | Number_of_employees |
+----------------+----------------+---------------------+
|              1 | ProductX       |                   3 |
|              2 | ProductY       |                   3 |
|             20 | Reorganization |                   3 |
|             30 | NewBenefits    |                   3 |
+----------------+----------------+---------------------+
4 rows in set (0.00 sec)
```

Query 27. For each project, retrieve the project number, the project name, and the number of employees from department 5 who work on the project.

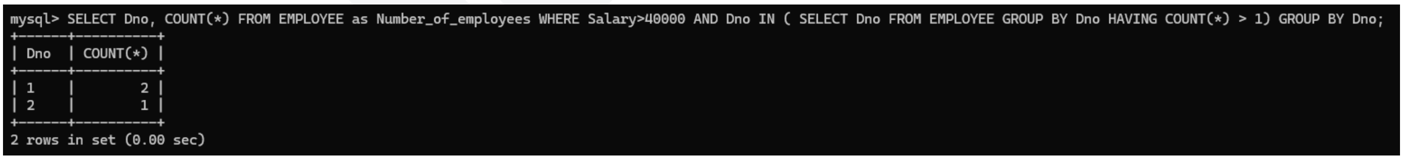
Output:



```
mysql>  SELECT Pnumber as Project_Number, Pname as Project_Name, COUNT(*) as Number_of_employees FROM PROJECT, WORKS_ON, EMPLOYEE WHERE Pnumber = Pno AND Ssn = Essn AND Dno = 5 GROUP BY Pnumber, Pname;
+----------------+-----------------+---------------------+
| Project_Number | Project_Name    | Number_of_employees |
+----------------+-----------------+---------------------+
|              1 | ProductX        |                   1 |
|              2 | ProductY        |                   2 |
|              3 | ProductZ        |                   1 |
|             10 | Computerization |                   1 |
|             20 | Reorganization  |                   1 |
|             30 | NewBenefits     |                   1 |
|             40 | ProductH        |                   1 |
+----------------+-----------------+---------------------+
7 rows in set (0.00 sec)
```

Query 28. For each department that has more than One employees, retrieve the department number and the number of its employees who are making more than $40,000.

Output:



```
mysql> SELECT Dno, COUNT(*) FROM EMPLOYEE as Number_of_employees WHERE Salary>40000 AND Dno IN ( SELECT Dno FROM EMPLOYEE GROUP BY Dno HAVING COUNT(*) > 1) GROUP BY Dno;
+------+----------+
| Dno  | COUNT(*) |
+------+----------+
| 1    |        2 |
| 2    |        1 |
+------+----------+
2 rows in set (0.00 sec)
```