Assignment No. 4

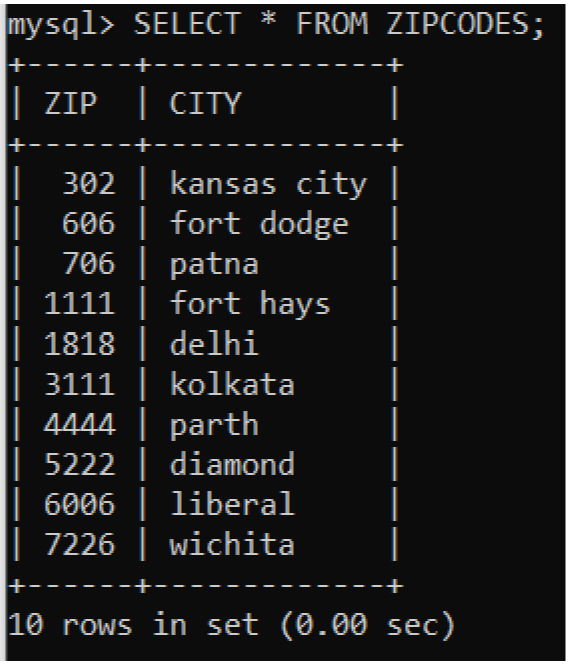
By
Neeraj Pratap Hazarika
2020csb040

A. Creation of Tables:

Creation of ZIPCODES table:

Query:

```
CREATE TABLE ZIPCODES(
ZIP INT NOT NULL,
CITY VARCHAR(100) NOT NULL,
PRIMARY KEY(ZIP)
);
```



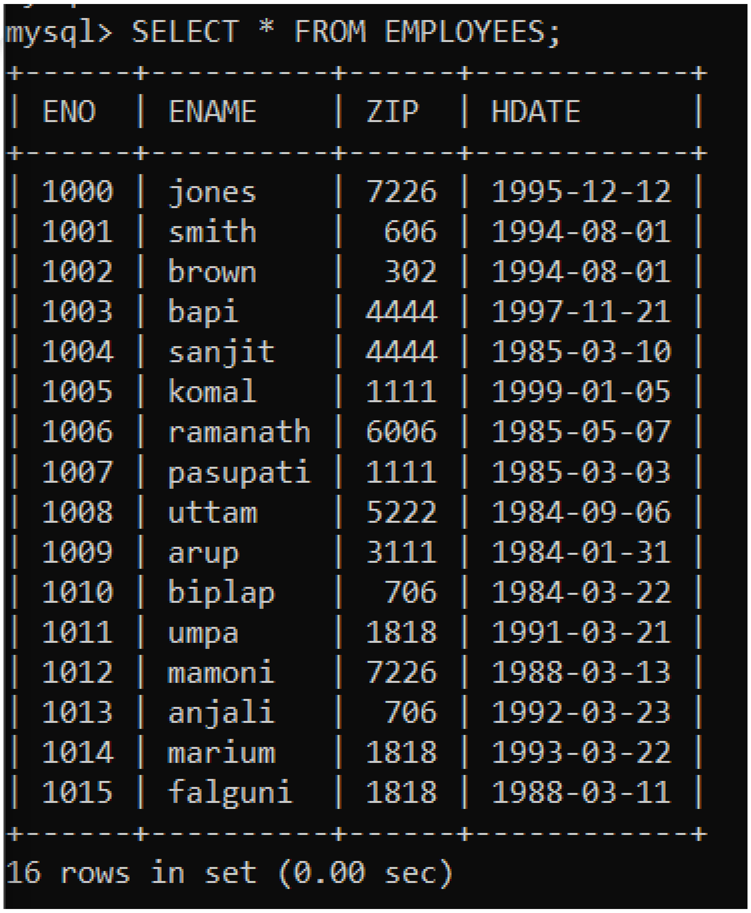
```
mysql> SELECT * FROM ZIPCODES;
+------+-------------+
| ZIP  | CITY        |
+------+-------------+
|  302 | kansas city |
|  606 | fort dodge  |
|  706 | patna       |
| 1111 | fort hays   |
| 1818 | delhi       |
| 3111 | kolkata     |
| 4444 | parth       |
| 5222 | diamond     |
| 6006 | liberal     |
| 7226 | wichita     |
+------+-------------+
10 rows in set (0.00 sec)
```

Creation of EMPLOYEES table:

Query:

```
CREATE TABLE EMPLOYEES(
ENO INT NOT NULL,
ENAME VARCHAR(100) NOT NULL,
ZIP INT NOT NULL,
HDATE DATE NOT NULL,
PRIMARY KEY(ENO),
FOREIGN KEY(ZIP) REFERENCES ZIPCODES(ZIP) ON DELETE
```

);

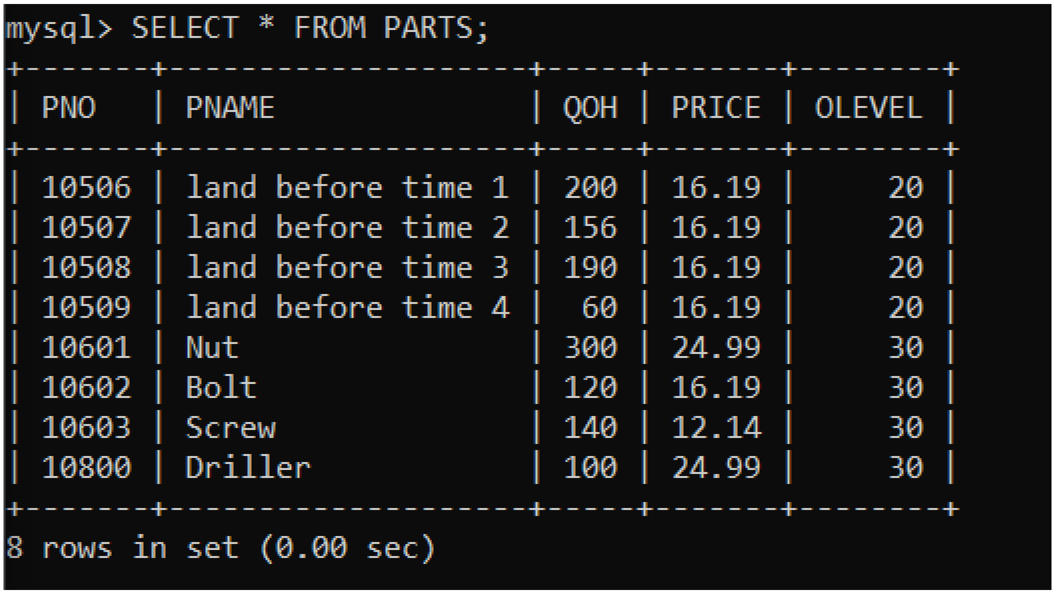


```
mysql> SELECT * FROM EMPLOYEES;
+------+----------+------+------------+
| ENO  | ENAME    | ZIP  | HDATE      |
+------+----------+------+------------+
| 1000 | jones    | 7226 | 1995-12-12 |
| 1001 | smith    |  606 | 1994-08-01 |
| 1002 | brown    |  302 | 1994-08-01 |
| 1003 | bapi     | 4444 | 1997-11-21 |
| 1004 | sanjit   | 4444 | 1985-03-10 |
| 1005 | komal    | 1111 | 1999-01-05 |
| 1006 | ramanath | 6006 | 1985-05-07 |
| 1007 | pasupati | 1111 | 1985-03-03 |
| 1008 | uttam    | 5222 | 1984-09-06 |
| 1009 | arup     | 3111 | 1984-01-31 |
| 1010 | biplap   |  706 | 1984-03-22 |
| 1011 | umpa     | 1818 | 1991-03-21 |
| 1012 | mamoni   | 7226 | 1988-03-13 |
| 1013 | anjali   |  706 | 1992-03-23 |
| 1014 | marium   | 1818 | 1993-03-22 |
| 1015 | falguni  | 1818 | 1988-03-11 |
+------+----------+------+------------+
16 rows in set (0.00 sec)
```

Creation of PARTS table:

Query:

```
CREATE TABLE PARTS(
PNO INT NOT NULL,
PNAME VARCHAR(100) NOT NULL,
QOH INT NOT NULL,
PRICE FLOAT(4,2) NOT NULL,
OLEVEL INT NOT NULL,
PRIMARY KEY(PNO)
);
```

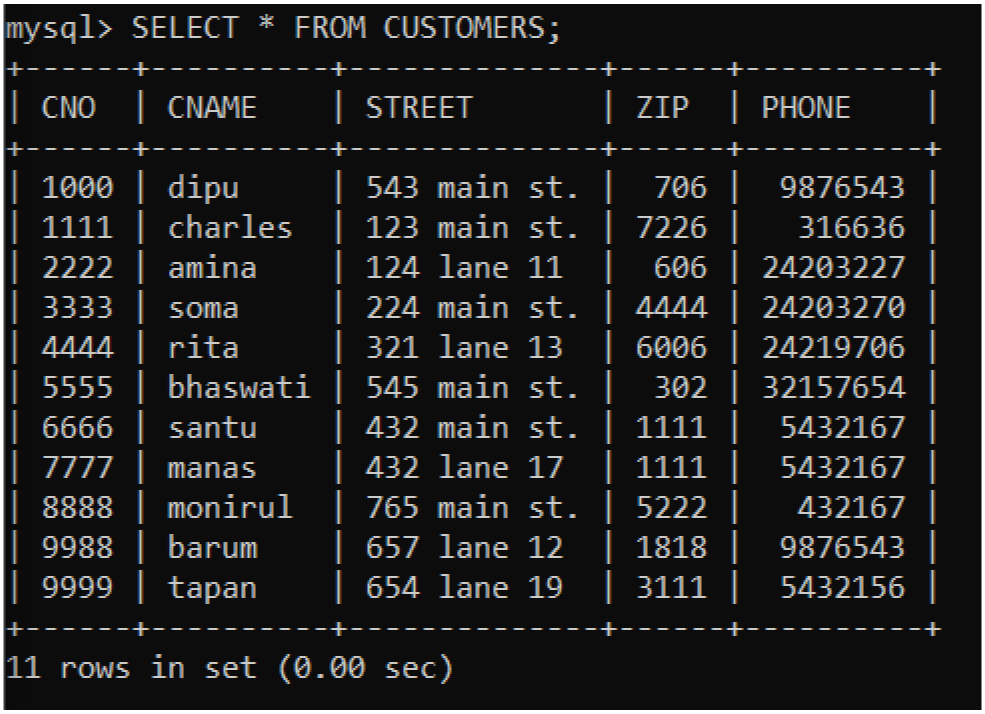


```
mysql> SELECT * FROM PARTS;
+-------+--------------------+-----+-------+--------+
| PNO   | PNAME              | QOH | PRICE | OLEVEL |
+-------+--------------------+-----+-------+--------+
| 10506 | land before time 1 | 200 | 16.19 |     20 |
| 10507 | land before time 2 | 156 | 16.19 |     20 |
| 10508 | land before time 3 | 190 | 16.19 |     20 |
| 10509 | land before time 4 |  60 | 16.19 |     20 |
| 10601 | Nut                | 300 | 24.99 |     30 |
| 10602 | Bolt               | 120 | 16.19 |     30 |
| 10603 | Screw              | 140 | 12.14 |     30 |
| 10800 | Driller            | 100 | 24.99 |     30 |
+-------+--------------------+-----+-------+--------+
8 rows in set (0.00 sec)
```

<u>Creation of CUSTOMERS table:</u>

Query:

```
CREATE TABLE CUSTOMERS(
CNO INT NOT NULL,
CNAME VARCHAR(100) NOT NULL,
STREET VARCHAR(100) NOT NULL,
ZIP INT NOT NULL,
PHONE INT NOT NULL,
PRIMARY KEY(CNO),
FOREIGN KEY(ZIP) REFERENCES ZIPCODES(ZIP)
);
```

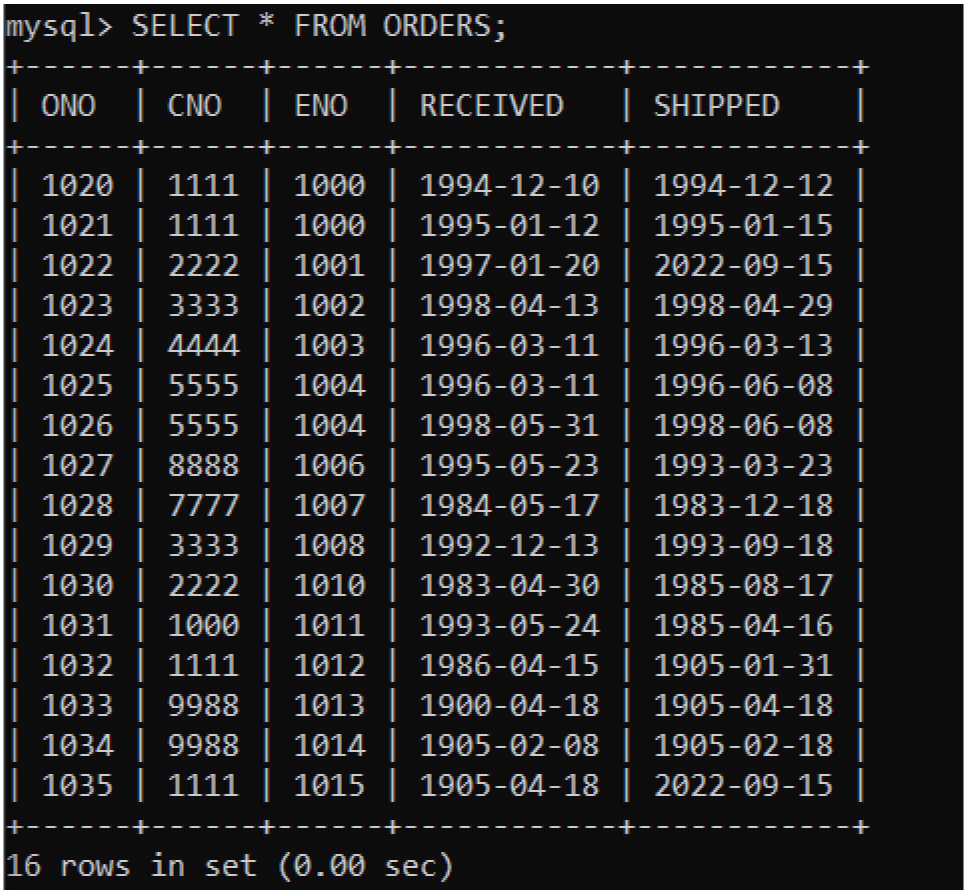


```
mysql> SELECT * FROM CUSTOMERS;
+------+----------+--------------+------+----------+
| CNO  | CNAME    | STREET       | ZIP  | PHONE    |
+------+----------+--------------+------+----------+
| 1000 | dipu     | 543 main st. |  706 |  9876543 |
| 1111 | charles  | 123 main st. | 7226 |   316636 |
| 2222 | amina    | 124 lane 11  |  606 | 24203227 |
| 3333 | soma     | 224 main st. | 4444 | 24203270 |
| 4444 | rita     | 321 lane 13  | 6006 | 24219706 |
| 5555 | bhaswati | 545 main st. |  302 | 32157654 |
| 6666 | santu    | 432 main st. | 1111 |  5432167 |
| 7777 | manas    | 432 lane 17  | 1111 |  5432167 |
| 8888 | monirul  | 765 main st. | 5222 |   432167 |
| 9988 | barum    | 657 lane 12  | 1818 |  9876543 |
| 9999 | tapan    | 654 lane 19  | 3111 |  5432156 |
+------+----------+--------------+------+----------+
11 rows in set (0.00 sec)
```

Creation of ORDERS table:

Query:

```
CREATE TABLE ORDERS(
ONO INT NOT NULL,
CNO INT NOT NULL,
ENO INT NOT NULL,
RECEIVED DATE NOT NULL,
SHIPPED DATE,
PRIMARY KEY(ONO),
FOREIGN KEY(CNO) REFERENCES CUSTOMERS(CNO),
FOREIGN KEY(ENO) REFERENCES EMPLOYEES(ENO)
);
```

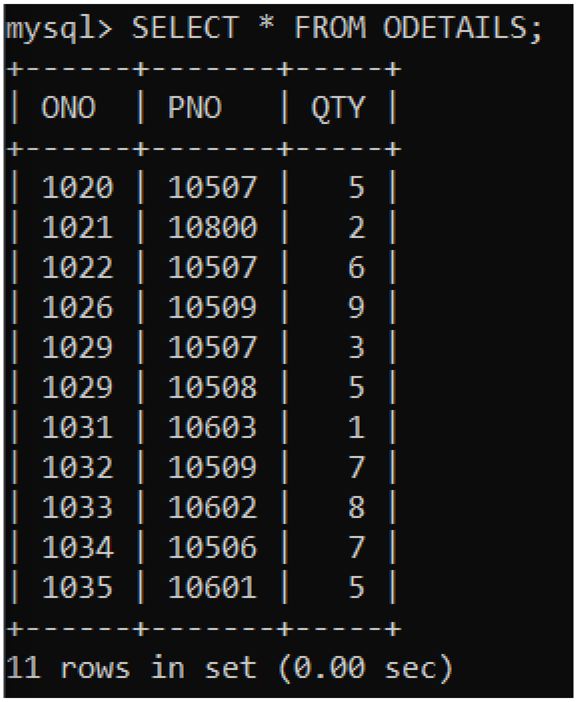


```
mysql> SELECT * FROM ORDERS;
+------+------+------+------------+------------+
| ONO  | CNO  | ENO  | RECEIVED   | SHIPPED    |
+------+------+------+------------+------------+
| 1020 | 1111 | 1000 | 1994-12-10 | 1994-12-12 |
| 1021 | 1111 | 1000 | 1995-01-12 | 1995-01-15 |
| 1022 | 2222 | 1001 | 1997-01-20 | 2022-09-15 |
| 1023 | 3333 | 1002 | 1998-04-13 | 1998-04-29 |
| 1024 | 4444 | 1003 | 1996-03-11 | 1996-03-13 |
| 1025 | 5555 | 1004 | 1996-03-11 | 1996-06-08 |
| 1026 | 5555 | 1004 | 1998-05-31 | 1998-06-08 |
| 1027 | 8888 | 1006 | 1995-05-23 | 1993-03-23 |
| 1028 | 7777 | 1007 | 1984-05-17 | 1983-12-18 |
| 1029 | 3333 | 1008 | 1992-12-13 | 1993-09-18 |
| 1030 | 2222 | 1010 | 1983-04-30 | 1985-08-17 |
| 1031 | 1000 | 1011 | 1993-05-24 | 1985-04-16 |
| 1032 | 1111 | 1012 | 1986-04-15 | 1905-01-31 |
| 1033 | 9988 | 1013 | 1900-04-18 | 1905-04-18 |
| 1034 | 9988 | 1014 | 1905-02-08 | 1905-02-18 |
| 1035 | 1111 | 1015 | 1905-04-18 | 2022-09-15 |
+------+------+------+------------+------------+
16 rows in set (0.00 sec)
```

Creation of ODETAILS table:

Query:

```
CREATE TABLE ODETAILS(
ONO INT NOT NULL,
PNO INT NOT NULL,
QTY INT NOT NULL,
PRIMARY KEY(ONO,PNO),
FOREIGN KEY(ONO) REFERENCES ORDERS(ONO),
FOREIGN KEY(PNO) REFERENCES PARTS(PNO)
);
```

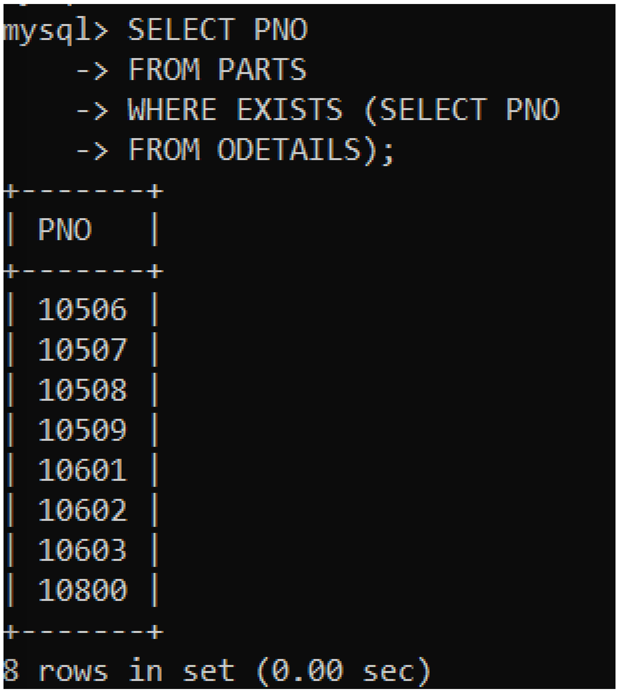


```
mysql> SELECT * FROM ODETAILS;
+------+-------+-----+
| ONO  | PNO   | QTY |
+------+-------+-----+
| 1020 | 10507 |   5 |
| 1021 | 10800 |   2 |
| 1022 | 10507 |   6 |
| 1026 | 10509 |   9 |
| 1029 | 10507 |   3 |
| 1029 | 10508 |   5 |
| 1031 | 10603 |   1 |
| 1032 | 10509 |   7 |
| 1033 | 10602 |   8 |
| 1034 | 10506 |   7 |
| 1035 | 10601 |   5 |
+------+-------+-----+
11 rows in set (0.00 sec)
```

B. Queries and their Solutions:

1. Get PNO values for PARTS for witch ORDERS have been placed.

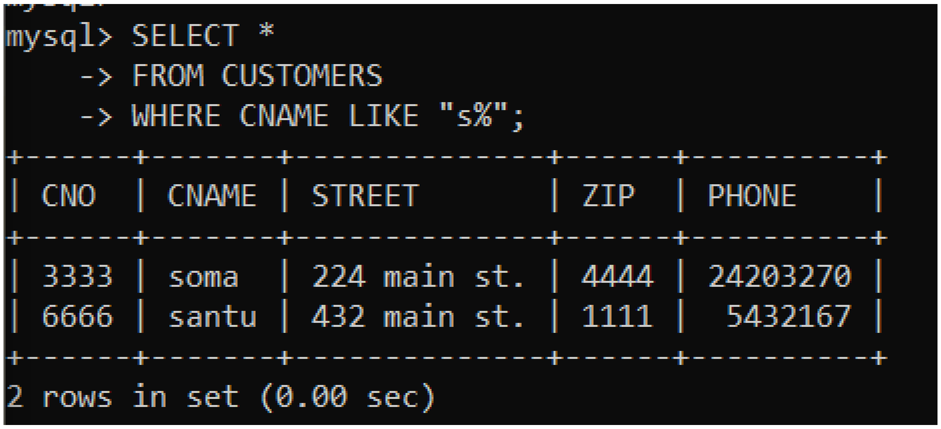
Query & Output:



```
mysql> SELECT PNO
    -> FROM PARTS
    -> WHERE EXISTS (SELECT PNO
    -> FROM ODETAILS);
+-------+
| PNO   |
+-------+
| 10506 |
| 10507 |
| 10508 |
| 10509 |
| 10601 |
| 10602 |
| 10603 |
| 10800 |
+-------+
8 rows in set (0.00 sec)
```

2. Get all the details of CUSTOMERS whose name has being letter ‘s’.

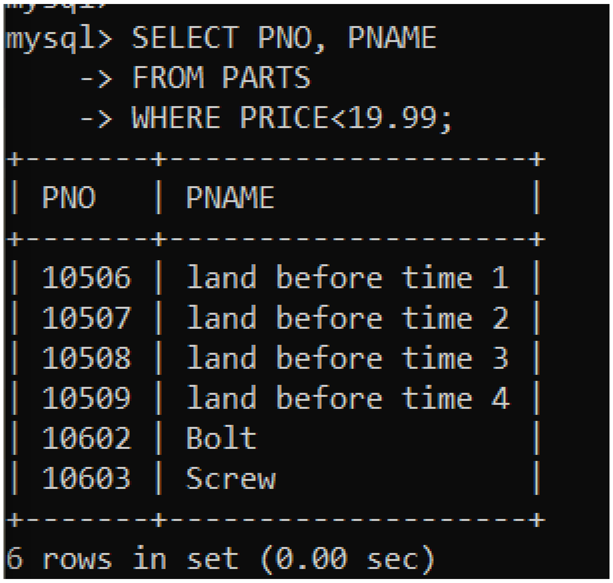
Query & Output:



```
mysql> SELECT *
    -> FROM CUSTOMERS
    -> WHERE CNAME LIKE "s%";
+------+-------+--------------+------+----------+
| CNO  | CNAME | STREET       | ZIP  | PHONE    |
+------+-------+--------------+------+----------+
| 3333 | soma  | 224 main st. | 4444 | 24203270 |
| 6666 | santu | 432 main st. | 1111 |  5432167 |
+------+-------+--------------+------+----------+
2 rows in set (0.00 sec)
```

3. Get PNO and PNAME values of PARTS that are priced less than 19.99.

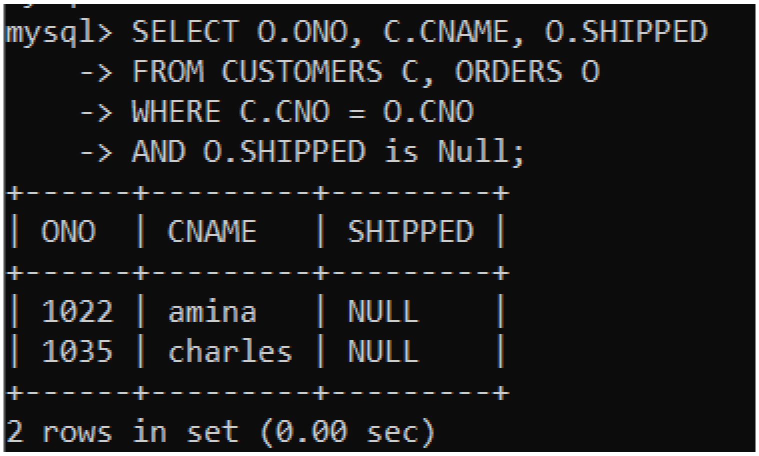
Query & Output:



```
mysql> SELECT PNO, PNAME
    -> FROM PARTS
    -> WHERE PRICE<19.99;
+-------+--------------------+
| PNO   | PNAME              |
+-------+--------------------+
| 10506 | land before time 1 |
| 10507 | land before time 2 |
| 10508 | land before time 3 |
| 10509 | land before time 4 |
| 10602 | Bolt               |
| 10603 | Screw              |
+-------+--------------------+
6 rows in set (0.00 sec)
```

4. Get the ONO, CNAME and SHIPPED values for CUSTOMERS whose orders have not yet been shipped.

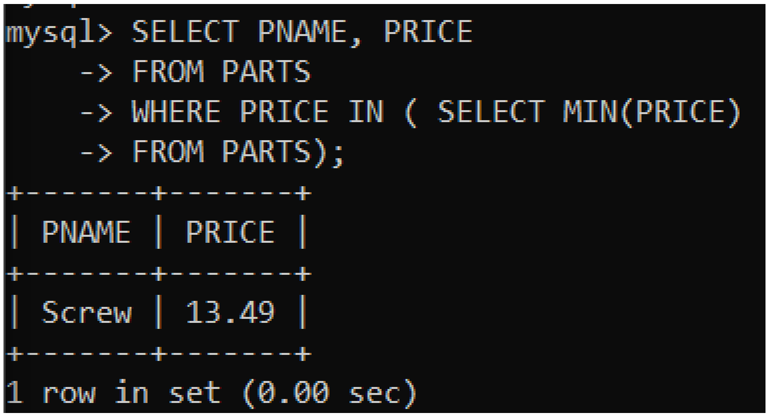
Query & Output:



```
mysql> SELECT O.ONO, C.CNAME, O.SHIPPED
    -> FROM CUSTOMERS C, ORDERS O
    -> WHERE C.CNO = O.CNO
    -> AND O.SHIPPED is Null;
+------+---------+---------+
| ONO  | CNAME   | SHIPPED |
+------+---------+---------+
| 1022 | amina   | NULL    |
| 1035 | charles | NULL    |
+------+---------+---------+
2 rows in set (0.00 sec)
```

5. Get PNAME and PRICE value from PARTS with the lowest PRICE.

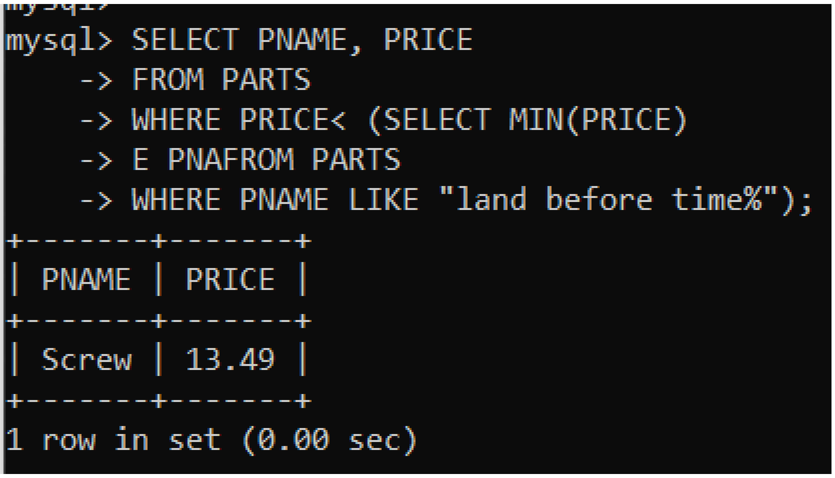
Query & Output:



```
mysql> SELECT PNAME, PRICE
    -> FROM PARTS
    -> WHERE PRICE IN ( SELECT MIN(PRICE)
    -> FROM PARTS);
+-------+-------+
| PNAME | PRICE |
+-------+-------+
| Screw | 13.49 |
+-------+-------+
1 row in set (0.00 sec)
```

6. Get the PNAME and PRICE values of PARTS that cost less than the least expensive ‘land before time’ part.

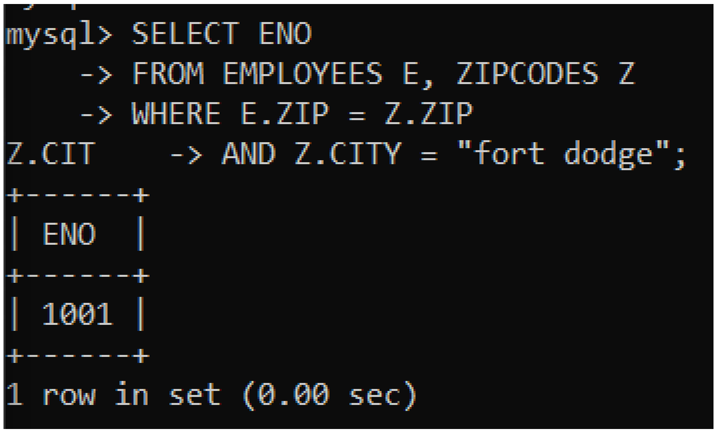
Query & Output:



```
mysql> SELECT PNAME, PRICE
    -> FROM PARTS
    -> WHERE PRICE< (SELECT MIN(PRICE)
    -> E PNAFROM PARTS
    -> WHERE PNAME LIKE "land before time%");
+-------+-------+
| PNAME | PRICE |
+-------+-------+
| Screw | 13.49 |
+-------+-------+
1 row in set (0.00 sec)
```

7. Get the ENO values of EMPLOYEES from ‘Fort Dodge’.

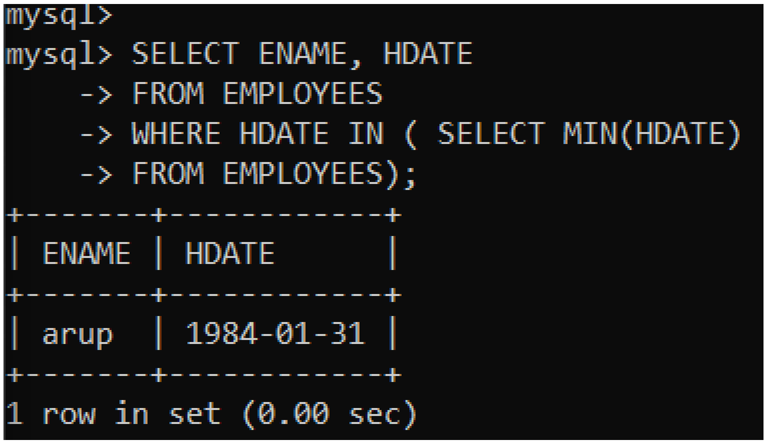
Query & Output:



```
mysql> SELECT ENO
    -> FROM EMPLOYEES E, ZIPCODES Z
    -> WHERE E.ZIP = Z.ZIP
Z.CIT    -> AND Z.CITY = "fort dodge";
+------+
| ENO  |
+------+
| 1001 |
+------+
1 row in set (0.00 sec)
```

8. Get the ENAME and HDATE of the EMPLOYEES who was hired on the earliest date.

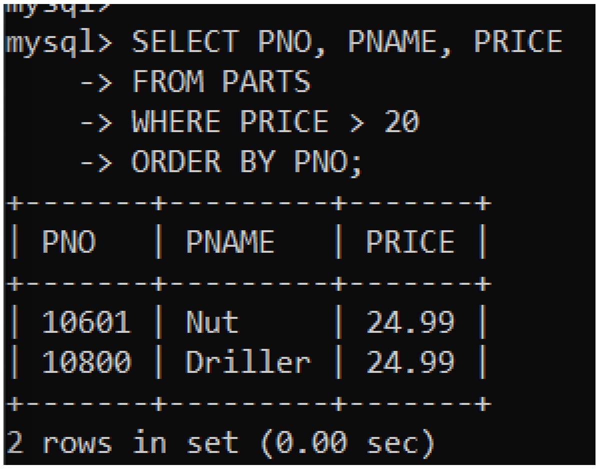
Query & Output:



```
mysql>
mysql> SELECT ENAME, HDATE
    -> FROM EMPLOYEES
    -> WHERE HDATE IN ( SELECT MIN(HDATE)
    -> FROM EMPLOYEES);
+-------+------------+
| ENAME | HDATE      |
+-------+------------+
| arup  | 1984-01-31 |
+-------+------------+
1 row in set (0.00 sec)
```

9. Retrieve the PNO, PNAME and PRICE of PARTS with price greater than 20.00 in an ascending order of PNO.

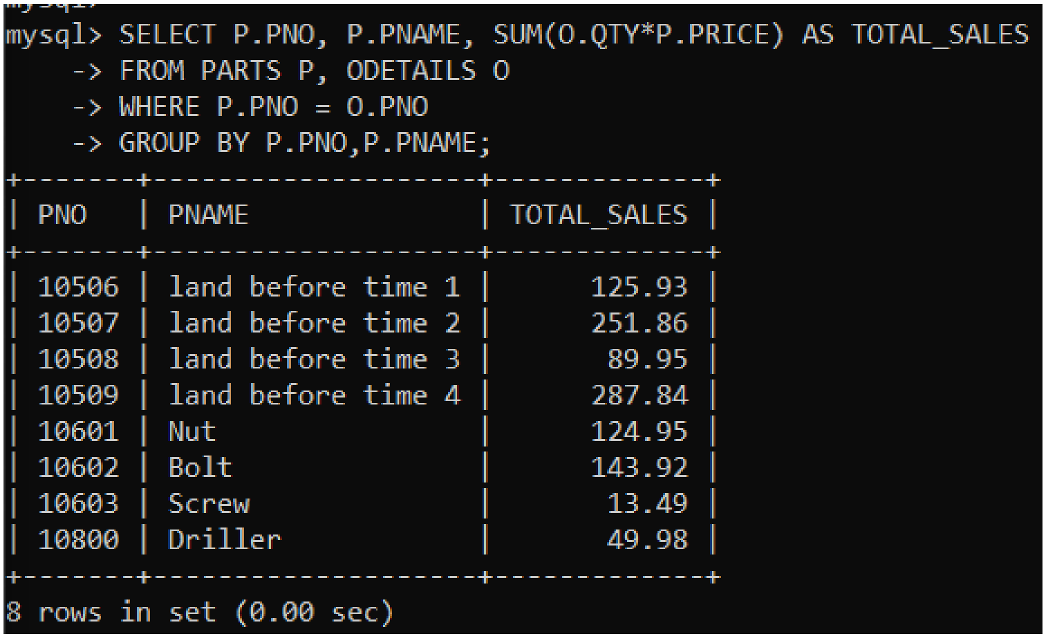
Query & Output:



```
mysql>
mysql> SELECT PNO, PNAME, PRICE
    -> FROM PARTS
    -> WHERE PRICE > 20
    -> ORDER BY PNO;
+-------+---------+-------+
| PNO   | PNAME   | PRICE |
+-------+---------+-------+
| 10601 | Nut     | 24.99 |
| 10800 | Driller | 24.99 |
+-------+---------+-------+
2 rows in set (0.00 sec)
```

10. For each PARTS get PNO and PNAME values along with total sales in details.

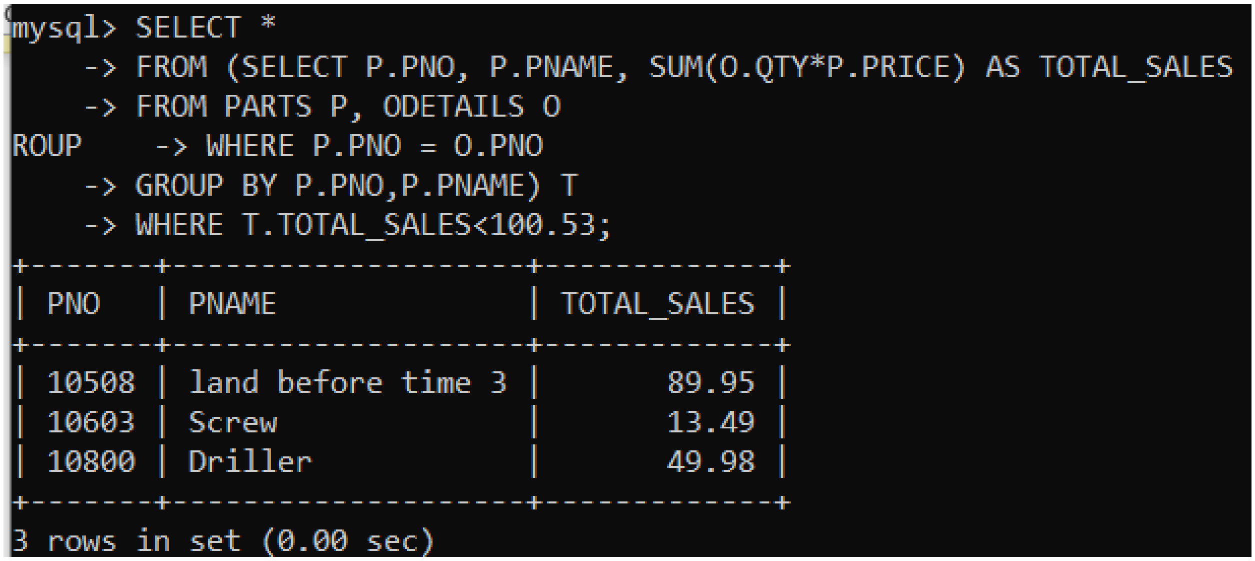
Query & Output:



```
mysql> SELECT P.PNO, P.PNAME, SUM(O.QTY*P.PRICE) AS TOTAL_SALES
    -> FROM PARTS P, ODETAILS O
    -> WHERE P.PNO = O.PNO
    -> GROUP BY P.PNO,P.PNAME;
+-------+--------------------+-------------+
| PNO   | PNAME              | TOTAL_SALES |
+-------+--------------------+-------------+
| 10506 | land before time 1 |      125.93 |
| 10507 | land before time 2 |      251.86 |
| 10508 | land before time 3 |       89.95 |
| 10509 | land before time 4 |      287.84 |
| 10601 | Nut                |      124.95 |
| 10602 | Bolt               |      143.92 |
| 10603 | Screw              |       13.49 |
| 10800 | Driller            |       49.98 |
+-------+--------------------+-------------+
8 rows in set (0.00 sec)
```

11. For each PARTS get PNO and PNAME values along with total sales in rupees but only when the total sales exceeds 100.53.

Query & Output:

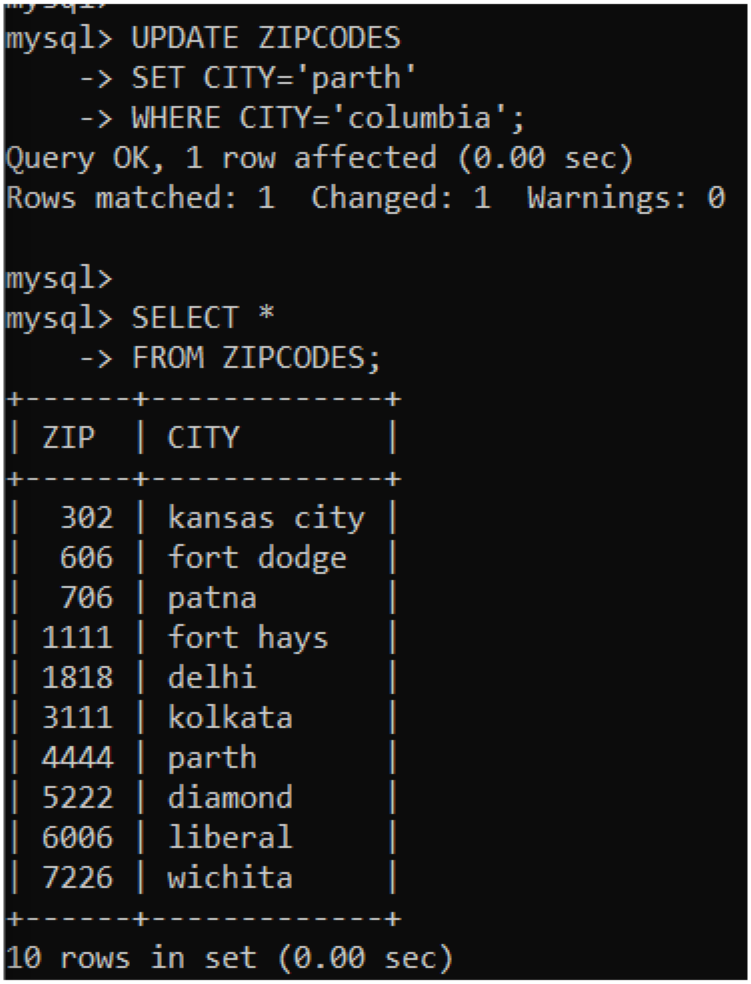


```
mysql> SELECT *
    -> FROM (SELECT P.PNO, P.PNAME, SUM(O.QTY*P.PRICE) AS TOTAL_SALES
    -> FROM PARTS P, ODETAILS O
ROUP    -> WHERE P.PNO = O.PNO
    -> GROUP BY P.PNO,P.PNAME) T
    -> WHERE T.TOTAL_SALES<100.53;
+-------+--------------------+-------------+
| PNO   | PNAME              | TOTAL_SALES |
+-------+--------------------+-------------+
| 10508 | land before time 3 |       89.95 |
| 10603 | Screw              |       13.49 |
| 10800 | Driller            |       49.98 |
+-------+--------------------+-------------+
3 rows in set (0.00 sec)
```

Correction: Query would “WHERE T.TOTAL_SALES<100.53;”.

12. Change the name of the CITY ‘columbia’ to ‘parth’.

Query & Output:

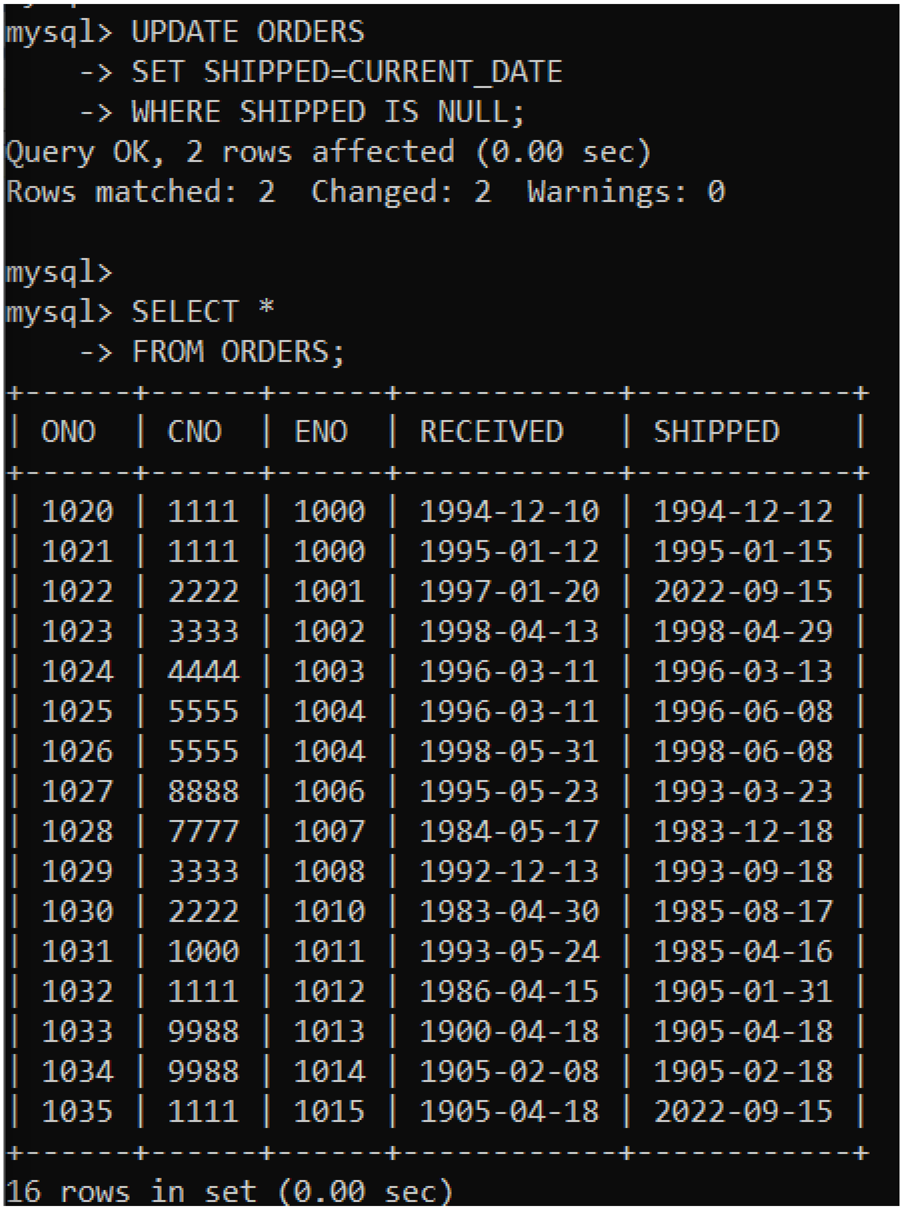


```
mysql>
mysql> UPDATE ZIPCODES
    -> SET CITY='parth'
    -> WHERE CITY='columbia';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
mysql> SELECT *
    -> FROM ZIPCODES;
+------+-------------+
| ZIP  | CITY        |
+------+-------------+
|  302 | kansas city |
|  606 | fort dodge  |
|  706 | patna       |
| 1111 | fort hays   |
| 1818 | delhi       |
| 3111 | kolkata     |
| 4444 | parth       |
| 5222 | diamond     |
| 6006 | liberal     |
| 7226 | wichita     |
+------+-------------+
10 rows in set (0.00 sec)
```

13. Update all the null valued SHIPPED dates to the current date.

Query & Output:

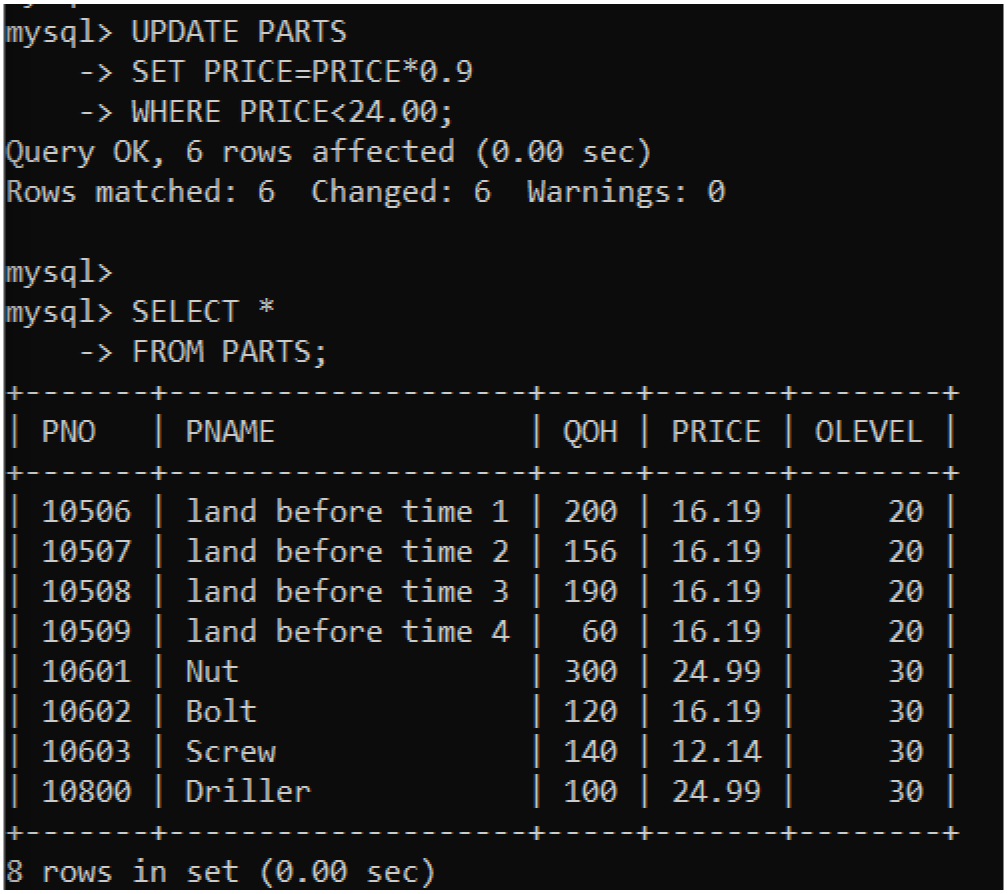


```
mysql> UPDATE ORDERS
    -> SET SHIPPED=CURRENT_DATE
    -> WHERE SHIPPED IS NULL;
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql>
mysql> SELECT *
    -> FROM ORDERS;
+------+------+------+------------+------------+
| ONO  | CNO  | ENO  | RECEIVED   | SHIPPED    |
+------+------+------+------------+------------+
| 1020 | 1111 | 1000 | 1994-12-10 | 1994-12-12 |
| 1021 | 1111 | 1000 | 1995-01-12 | 1995-01-15 |
| 1022 | 2222 | 1001 | 1997-01-20 | 2022-09-15 |
| 1023 | 3333 | 1002 | 1998-04-13 | 1998-04-29 |
| 1024 | 4444 | 1003 | 1996-03-11 | 1996-03-13 |
| 1025 | 5555 | 1004 | 1996-03-11 | 1996-06-08 |
| 1026 | 5555 | 1004 | 1998-05-31 | 1998-06-08 |
| 1027 | 8888 | 1006 | 1995-05-23 | 1993-03-23 |
| 1028 | 7777 | 1007 | 1984-05-17 | 1983-12-18 |
| 1029 | 3333 | 1008 | 1992-12-13 | 1993-09-18 |
| 1030 | 2222 | 1010 | 1983-04-30 | 1985-08-17 |
| 1031 | 1000 | 1011 | 1993-05-24 | 1985-04-16 |
| 1032 | 1111 | 1012 | 1986-04-15 | 1905-01-31 |
| 1033 | 9988 | 1013 | 1900-04-18 | 1905-04-18 |
| 1034 | 9988 | 1014 | 1905-02-08 | 1905-02-18 |
| 1035 | 1111 | 1015 | 1905-04-18 | 2022-09-15 |
+------+------+------+------------+------------+
16 rows in set (0.00 sec)
```

14. Decrease the PRICE of all PARTS that cost less than 24.00 by 10%.

Query & Output:

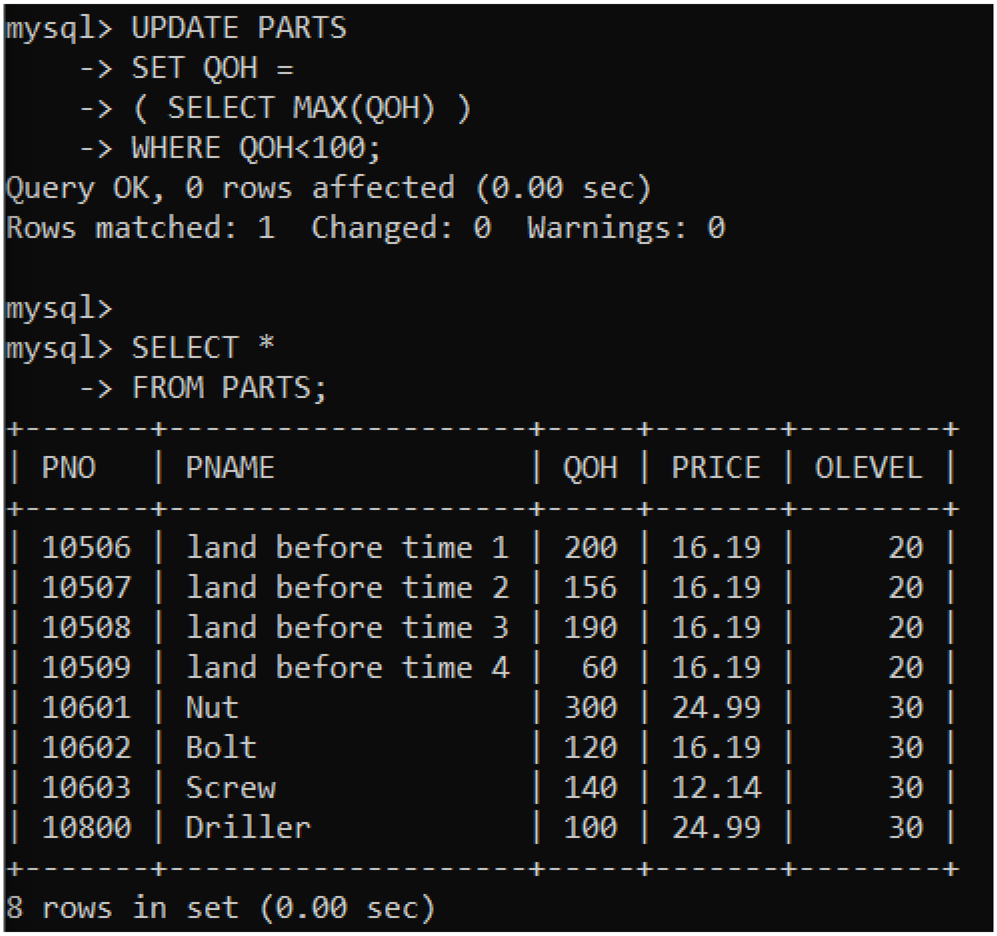


```
mysql> UPDATE PARTS
    -> SET PRICE=PRICE*0.9
    -> WHERE PRICE<24.00;
Query OK, 6 rows affected (0.00 sec)
Rows matched: 6  Changed: 6  Warnings: 0

mysql>
mysql> SELECT *
    -> FROM PARTS;
+-------+--------------------+-----+-------+--------+
| PNO   | PNAME              | QOH | PRICE | OLEVEL |
+-------+--------------------+-----+-------+--------+
| 10506 | land before time 1 | 200 | 16.19 |     20 |
| 10507 | land before time 2 | 156 | 16.19 |     20 |
| 10508 | land before time 3 | 190 | 16.19 |     20 |
| 10509 | land before time 4 |  60 | 16.19 |     20 |
| 10601 | Nut                | 300 | 24.99 |     30 |
| 10602 | Bolt               | 120 | 16.19 |     30 |
| 10603 | Screw              | 140 | 12.14 |     30 |
| 10800 | Driller            | 100 | 24.99 |     30 |
+-------+--------------------+-----+-------+--------+
8 rows in set (0.00 sec)
```

15. Set the ‘QOH’ value of those PARTS whose current ‘QOH’ value is less than 100 to the maximum ‘QOH’ value present in the table.

Query & Output:



```
mysql> UPDATE PARTS
    -> SET QOH =
    -> ( SELECT MAX(QOH) )
    -> WHERE QOH<100;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1  Changed: 0  Warnings: 0

mysql>
mysql> SELECT *
    -> FROM PARTS;
+-------+--------------------+-----+-------+--------+
| PNO   | PNAME              | QOH | PRICE | OLEVEL |
+-------+--------------------+-----+-------+--------+
| 10506 | land before time 1 | 200 | 16.19 |     20 |
| 10507 | land before time 2 | 156 | 16.19 |     20 |
| 10508 | land before time 3 | 190 | 16.19 |     20 |
| 10509 | land before time 4 |  60 | 16.19 |     20 |
| 10601 | Nut                | 300 | 24.99 |     30 |
| 10602 | Bolt               | 120 | 16.19 |     30 |
| 10603 | Screw              | 140 | 12.14 |     30 |
| 10800 | Driller            | 100 | 24.99 |     30 |
+-------+--------------------+-----+-------+--------+
8 rows in set (0.00 sec)
```