ASSIGNMENT-4

NAME-ANUBHAV ANAND

ENROLLMENT NUMBER-2020CSB102

SUBJECT- ASSIGNMENT 4 OF DBMS

G-SUITE ID-2020CSB102.anubhav@students.iiests.ac.in

- **Creation of ZIPCODES table-**
- **Query for creating table:-**

```
->create table ZIPCODES
->(ZIP int(5) PRIMARY KEY,
->CITY varchar(30));
```

- **Description of ZIPCODES TABLE:-**

```
mysql> desc ZIPCODES;
+-------+------------+------+-----+---------+-------+
| Field | Type       | Null | Key | Default | Extra |
+-------+------------+------+-----+---------+-------+
| ZIP   | int        | NO   | PRI | NULL    |       |
| CITY  | varchar(30) | YES  |     | NULL    |       |
+-------+------------+------+-----+---------+-------+
2 rows in set (0.04 sec)
```

- **Query for inserting values into the table-**

```
->INSERT into ZIPCODES
->values(67226,'wichita');
```

- **ZIPCODES table after inserting values in it:-**

```
mysql> select * from ZIPCODES;
+-------+--------------+
| ZIP   | CITY         |
+-------+--------------+
| 50302 | kansas city  |
| 54444 | columbia     |
| 60606 | fort dodge   |
| 60706 | patna        |
| 61111 | fort hays    |
| 61818 | delhi        |
| 63111 | kolkata      |
| 65222 | diamond      |
| 66006 | libaral      |
| 67226 | wichita      |
+-------+--------------+
10 rows in set (0.00 sec)
```

- **Creating table EMPLOYEES:-**
- **Query for creating the table:-**

```
->create table EMPLOYEES
->(ENO int(4) PRIMARY KEY,
->ENAME varchar(30),
->ZIP int(5),
->HDATE date,
->constraint ZIP foreign key(ZIP)
->references ZIPCODES(ZIP) on delete
->cascade);
```

- **Description of Employees table-**

```
mysql> desc employees;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| ENO   | int         | NO   | PRI | NULL    |       |
| ENAME | varchar(30) | YES  |     | NULL    |       |
| ZIP   | int         | YES  | MUL | NULL    |       |
| HDATE | date        | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

- **Query for inserting values in EMPLOYEES Table-**

  ->INSERT into EMPLOYEES

  ->values

  ->(1000,'jones', 67226 ,'1995-12-12');

- **Table Employees after inserting values-**

MySQL 8.0 Command Line Client

```
+------+----------+-------+------------+
| ENO  | ENAME    | ZIP   | HDATE      |
+------+----------+-------+------------+
| 1000 | jones    | 67226 | 1995-12-12 |
| 1001 | smith    | 60606 | 1994-09-01 |
| 1002 | brown    | 50302 | 1994-09-01 |
| 1003 | bapi     | 54444 | 1997-11-21 |
| 1004 | sanjit   | 54444 | 1985-02-10 |
| 1005 | komal    | 61111 | 1999-01-05 |
| 1006 | ramanath | 66006 | 1985-05-07 |
| 1007 | pasupati | 61111 | 1985-03-03 |
| 1008 | uttam    | 65222 | 1984-09-06 |
| 1009 | arup     | 63111 | 1984-01-31 |
| 1010 | biplab   | 60706 | 1984-03-22 |
| 1011 | umpa     | 61818 | 1991-03-21 |
| 1012 | mamoni   | 67226 | 1988-03-13 |
| 1013 | anjali   | 60706 | 1992-03-22 |
| 1014 | marium   | 61818 | 1993-03-22 |
| 1015 | falguni  | 61818 | 1988-03-11 |
+------+----------+-------+------------+
16 rows in set (0.00 sec)
```

- **Creating table PARTS-**
- **Query for creating table-**

```
->CREATE table PARTS
->(PNO int(5) PRIMARY KEY,
->PNAME varchar(30),
->QOH int(3),
->PRICE double(6,3),
->OLEVEL int(2));
```

- **Description of PARTS Table-**

```
mysql> desc parts;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| PNO    | int         | NO   | PRI | NULL    |       |
| PNAME  | varchar(30) | YES  |     | NULL    |       |
| QOH    | int         | YES  |     | NULL    |       |
| PRICE  | double(6,3) | YES  |     | NULL    |       |
| OLEVEL | int         | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

- **Query for inserting values into parts-**

  ->INSERT into PARTS
  ->values
  ->(10506, 'land before time 1', 200, 17.99, 20);

- **TABLE PARTS after inserting all values-**

```
mysql> select * from parts;
+-------+--------------------+------+--------+--------+
| PNO   | PNAME              | QOH  | PRICE  | OLEVEL |
+-------+--------------------+------+--------+--------+
| 10506 | land before time 1 |  200 | 17.990 |     20 |
| 10507 | land before time 2 |  156 | 17.990 |     20 |
| 10508 | land before time 3 |  190 | 17.990 |     20 |
| 10509 | land before time 4 |   60 | 17.990 |     20 |
| 10601 | Nut                |  300 | 24.990 |     30 |
| 10602 | Bolt               |  120 | 17.990 |     30 |
| 10603 | Screw              |  140 | 13.490 |     30 |
| 10800 | Driller            |  100 | 24.990 |     30 |
+-------+--------------------+------+--------+--------+
8 rows in set (0.00 sec)
```

- **Creating table customers-**
- **Query for creating table customers-**

```
->CREATE table CUSTOMERS
->(CNO int(4) PRIMARY KEY,
->CNAME varchar(30),
->STREET varchar(30),
->ZIP int(5),
->PHONE int(10),
->constraint Z foreign key(ZIP) references
->ZIPCODES(ZIP) on delete cascade);
```

- **Description of Customers Table-**

```
mysql> desc customers;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| CNO    | int         | NO   | PRI | NULL    |       |
| CNAME  | varchar(30) | YES  |     | NULL    |       |
| STREET | varchar(30) | YES  |     | NULL    |       |
| ZIP    | int         | YES  | MUL | NULL    |       |
| PHONE  | int         | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

- **Query for inserting values in Customers table-**

  ->INSERT into CUSTOMERS

  ->values

  ->(1111, 'charles', '123 main st.', 67226, 316636);

- **Customers table after insering values-**

```
mysql> select * from customers;
+------+----------+--------------+-------+----------+
| CNO  | CNAME    | STREET       | ZIP   | PHONE    |
+------+----------+--------------+-------+----------+
| 1000 | dipu     | 543 main st. | 60706 |  9876543 |
| 1111 | charles  | 123 main st. | 67226 |   316636 |
| 2222 | amina    | 124 lane 11  | 60606 | 24203227 |
| 3333 | soma     | 224 main st. | 54444 | 24203270 |
| 4444 | rita     | 321 lane 13  | 66006 | 24219706 |
| 5555 | bhaswati | 543 main st. | 50302 | 32157654 |
| 6666 | santu    | 432 main st  | 61111 |  5432167 |
| 7777 | manas    | 432 lane 17  | 61111 |  5432167 |
| 8888 | monirul  | 765 main st. | 65222 |  4321566 |
| 9988 | barun    | 657 lane 12  | 61818 |  9876543 |
| 9999 | tapan    | 654 lane 19  | 63111 |  5432156 |
+------+----------+--------------+-------+----------+
11 rows in set (0.00 sec)
```

- **Creating table orders-**

```
->CREATE table ORDERS
->(ONO int(4) PRIMARY KEY,
->CNO int(4),
->ENO int(4),
->RECIEVED date,
->SHIPPED date,
->constraint CNO foreign key(CNO) -
>references CUSTOMERS(CNO) on delete
->cascade,
->constraint ENO foreign key(ENO)
->references EMPLOYEES(ENO) on delete
->cascade);
```

- **Description of Orders table-**

```
mysql> desc orders;
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| ONO      | int  | NO   | PRI | NULL    |       |
| CNO      | int  | YES  | MUL | NULL    |       |
| ENO      | int  | YES  | MUL | NULL    |       |
| RECIEVED | date | YES  |     | NULL    |       |
| SHIPPED  | date | YES  |     | NULL    |       |
+----------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

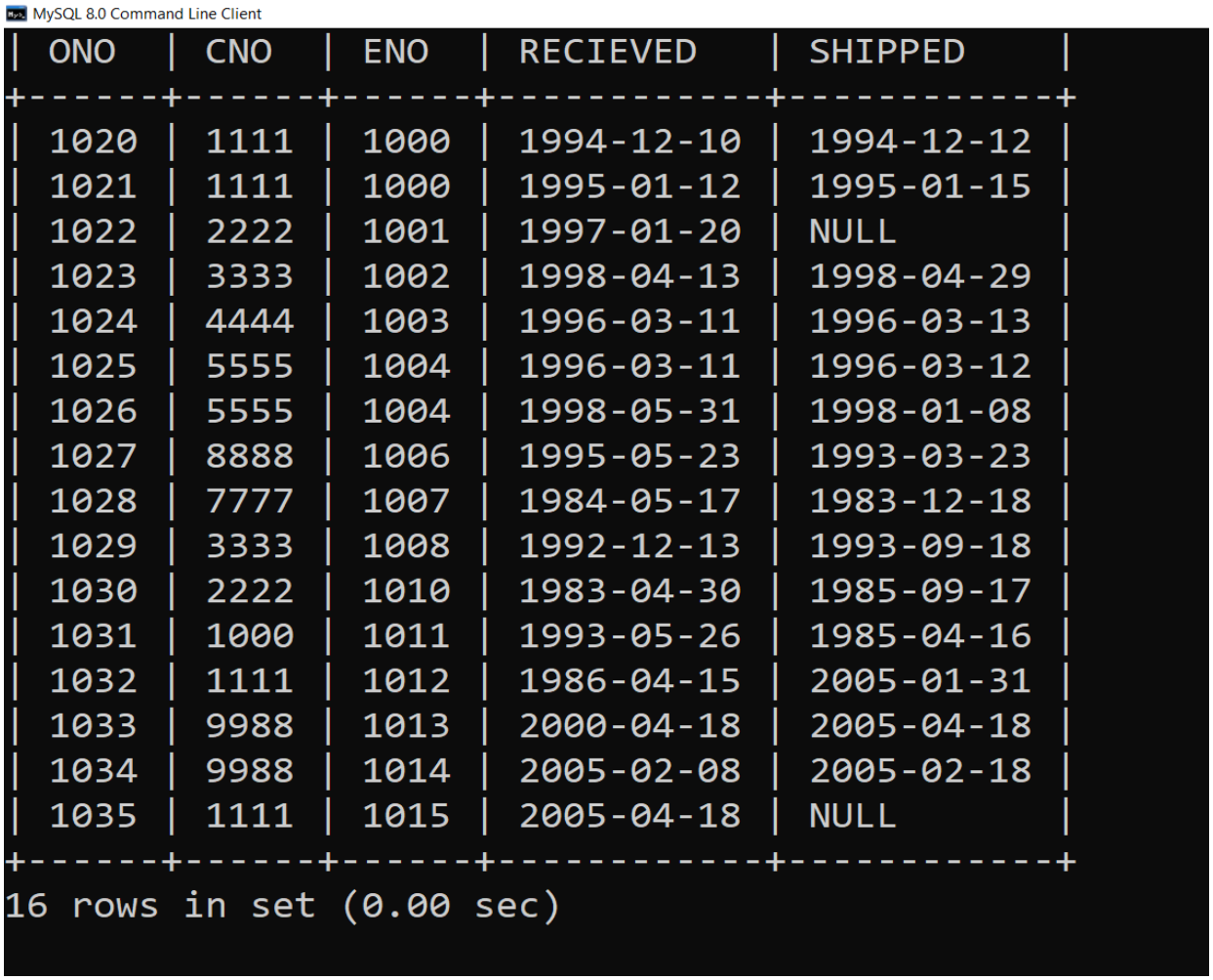
- **Inserting values into Orders-**

  ->INSERT into ORDERS

  ->values

  ->(1020, 1111, 1000, '1994-12-10', '1994-12-12');

- **Table orders after inserting values-**



- **Creating table ODETAILS-**
- **Query for creating table-**

  ->CREATE TABLE ODETAILS

  ->(ONO int(4),

  ->PNO int(5),

  ->QTY int(3),

  ->PRIMARY KEY(ONO,PNO),

  ->constraint ONO foreign key(ONO)

  ->references ORDERS(ONO) on delete

  ->cascade,

  ->constraint PNO foreign key(PNO)

  ->references PARTS(PNO) on delete

  ->cascade);

- **Description of ODETAILS-**

```
mysql> desc odetails;
+-------+------+------+-----+---------+-------+
| Field | Type | Null | Key | Default | Extra |
+-------+------+------+-----+---------+-------+
| ONO   | int  | NO   | PRI | NULL    |       |
| PNO   | int  | NO   | PRI | NULL    |       |
| QTY   | int  | YES  |     | NULL    |       |
+-------+------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

- **Query for inserting values in ODETAILS-**

  ->INSERT into ODETAILS

  ->values

  ->(1034, 10506, 7);

- **ODETAILS table after inserting values-**

```
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)
```

Answer the following Queries-

1.Get PNO values of PARTS for which orders have been placed.

Ans-Query-

mysql> select distinct PNO from ODETAILS;

```
mysql> select distict PNO from ODETAILS;
ERROR 1054 (42S22): Unknown column 'distict' in 'field list'
mysql> select distinct PNO from ODETAILS;
+-------+
| PNO   |
+-------+
| 10506 |
| 10507 |
| 10508 |
| 10509 |
| 10601 |
| 10602 |
| 10603 |
| 10800 |
+-------+
```

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

Ans-

Query-

```
mysql>
select * from customers where cname like
's%';
```

```
mysql> select * from customers where cname like 's%';
+------+-------+-------------+-------+----------+
| CNO  | CNAME | STREET      | ZIP   | PHONE    |
+------+-------+-------------+-------+----------+
| 3333 | soma  | 224 main st. | 54444 | 24203270 |
| 6666 | santu | 432 main st  | 61111 |  5432167 |
+------+-------+-------------+-------+----------+
2 rows in set (0.00 sec)
```

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

Ans-

Query-

```
mysql>
select PNO,PNAME from parts where price<19.99;
```

```
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)
```

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

Ans-

Query-

```
mysql> select ono,cname,shipped
    ->from customers,orders
    ->where customers.cno=orders.cno
    ->and orders.shipped is NULL;
```

We can also write It as-

```
mysql> select ono,cname,shipped
    ->from customers c,orders o
    ->where c.cno=o.cno
    ->and o.shipped is NULL;
```

```
mysql> select ono,cname,shipped
    ->  from customers,orders
    -> where customers.cno=orders.cno
    -> and orders.shipped is NULL;
+------+---------+---------+
| ono  | cname   | shipped |
+------+---------+---------+
| 1022 | amina   | NULL    |
| 1035 | charles | NULL    |
+------+---------+---------+
2 rows in set (0.00 sec)
```

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

Ans-

Query-

```
mysql> select PNAME,PRICE
    -> from PARTS
    -> order by PRICE
    -> limit 1;
```

```
mysql> select PNAME,PRICE
    -> from PARTS
    -> order by PRICE
    -> limit 1;
+-------+--------+
| PNAME | PRICE  |
+-------+--------+
| Screw | 13.490 |
+-------+--------+
1 row in set (0.00 sec)
```

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

Ans-

Query-

```
mysql> select PNAME,PRICE
   -> from PARTS where
   ->  PRICE<(
   ->  select min(PRICE)
   ->  FROM PARTS
   ->  where PNAME like 'land before time %');
```

```
mysql> select PNAME,PRICE
    -> from PARTS where
    ->  PRICE<(
    ->  select min(PRICE)
    ->  FROM PARTS
    ->  where PNAME like 'land before time %');
+-------+--------+
| PNAME | PRICE  |
+-------+--------+
| Screw | 13.490 |
+-------+--------+
1 row in set (0.00 sec)
```

7Q. Get the ENO values of EMPLOYEES from 'Fort Dodge'.

Ans-

Query-

```
mysql> select ENO
    -> from ZIPCODES,EMPLOYEES
    -> where ZIPCODES.ZIP=EMPLOYEES.ZIP
    -> and city='fort dodge';
```

We can also write it as-

```
mysql> select ENO
    -> from ZIPCODES z,EMPLOYEES e
    -> where z.ZIP=e.ZIP
    -> and city='fort dodge';
```

```
mysql> select ENO
    -> from ZIPCODES,EMPLOYEES
    -> where ZIPCODES.ZIP=EMPLOYEES.ZIP
    -> and city='fort dodge';
+------+
| ENO  |
+------+
| 1001 |
+------+
1 row in set (0.00 sec)
```

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

Ans-

Query-

```
mysql> select ENAME,HDATE
   -> from EMPLOYEES
   -> order by HDATE
   -> limit 1;
```

```
mysql> select ENAME,HDATE
    -> from EMPLOYEES
    -> order by HDATE
    -> limit 1;
+-------+------------+
| ENAME | HDATE      |
+-------+------------+
| arup  | 1984-01-31 |
+-------+------------+
1 row in set (0.00 sec)
```

9Q. Retrieve the PNO, PNAME and PRICE of PARTS with price greater than

20.00 in an ascending order of PNO.

Ans-Query-

```
mysql> select PNO,PNAME,PRICE
    -> from PARTS
    -> where PRICE>20.00
    -> order by PNO;
```

```
mysql> select PNO,PNAME,PRICE
    -> from PARTS
    -> where PRICE>20.00
    -> order by PNO;
+-------+---------+--------+
| PNO   | PNAME   | PRICE  |
+-------+---------+--------+
| 10601 | Nut     | 24.990 |
| 10800 | Driller | 24.990 |
+-------+---------+--------+
2 rows in set (0.00 sec)
```

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

details.

Ans-Query-

```
mysql>  select
PARTS.PNO,PARTS.PNAME,SUM(QTY*PRICE)
TOTAL_Sales
   ->  FROM PARTS,ODETAILS
   -> where PARTS.PNO=ODETAILS.PNO
   -> group by PNO;
```

We can write it as-

```
mysql>  select
P.PNO,P.PNAME,SUM(QTY*PRICE)
TOTAL_Sales
   ->  FROM PARTS P,ODETAILS O
   -> where P.PNO=O.PNO
   -> group by PNO;
```

```
mysql>  select PARTS.PNO,PARTS.PNAME,SUM(QTY*PRICE)
    ->  FROM PARTS,ODETAILS
    -> where PARTS.PNO=ODETAILS.PNO
    -> group by PNO;
+-------+--------------------+-------------+
| PNO   | PNAME              | TOTAL_Sales |
+-------+--------------------+-------------+
| 10506 | land before time 1 |     125.930 |
| 10507 | land before time 2 |     251.860 |
| 10508 | land before time 3 |      89.950 |
| 10509 | land before time 4 |     287.840 |
| 10601 | Nut                |     124.950 |
| 10602 | Bolt               |     143.920 |
| 10603 | Screw              |      13.490 |
| 10800 | Driller            |      49.980 |
+-------+--------------------+-------------+
8 rows in set (0.00 sec)
```

11Q. For each PARTS get PNO and PNAME values along with total sales in

rupees but only when the total sales exceeds 100.53.

Ans-Query-

```
mysql> select P.PNO,P.PNAME,concat('Rs. ',(sum(QTY*PRICE))) Total_Sales
    ->  from PARTS P,ODETAILS O
    ->  where P.PNO=O.PNO
    ->  group by PNO
    -> having sum(QTY*PRICE)>100.53;
+-------+--------------------+-------------+
| PNO   | PNAME              | Total_Sales |
+-------+--------------------+-------------+
| 10506 | land before time 1 | Rs. 125.930 |
| 10507 | land before time 2 | Rs. 251.860 |
| 10509 | land before time 4 | Rs. 287.840 |
| 10601 | Nut                | Rs. 124.950 |
| 10602 | Bolt               | Rs. 143.920 |
+-------+--------------------+-------------+
5 rows in set (0.01 sec)
```

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

Ans-Query-

```
mysql> update ZIPCODES set city='parth'
   -> where city='columbia';
```

Table Before Changing the city name-

```
mysql> select * from ZIPCODES;
+-------+-------------+
| ZIP   | CITY        |
+-------+-------------+
| 50302 | kansas city |
| 54444 | columbia    |
| 60606 | fort dodge  |
| 60706 | patna       |
| 61111 | fort hays   |
| 61818 | delhi       |
| 63111 | kolkata     |
| 65222 | diamond     |
| 66006 | libaral     |
| 67226 | wichita     |
+-------+-------------+
```

Table After Changing the city name-

ZIP	CITY
50302	kansas city
54444	parth
60606	fort dodge
60706	patna
61111	fort hays
61818	delhi
63111	kolkata
65222	diamond
66006	libaral
67226	wichita

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

Ans-Query-

```
mysql> update orders
   ->set shipped=SYSDATE()
   -> where shipped is null;
```

Before Changing The Table-

ONO	CNO	ENO	RECIEVED	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	NULL
1023	3333	1002	1998-04-13	1998-04-29
1024	4444	1003	1996-03-11	1996-03-13
1025	5555	1004	1996-03-11	1996-03-12
1026	5555	1004	1998-05-31	1998-01-08
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	1985-09-17
1031	1000	1011	1993-05-26	1985-04-16
1032	1111	1012	1986-04-15	2005-01-31
1033	9988	1013	2000-04-18	2005-04-18
1034	9988	1014	2005-02-08	2005-02-18
1035	1111	1015	2005-04-18	NULL

16 rows in set (0.00 sec)

After Changing the Table-

```
mysql> select * from ORDERS;
+------+------+------+------------+------------+
| ONO  | CNO  | ENO  | RECIEVED   | 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 | 2022-09-08 |
| 1023 | 3333 | 1002 | 1998-04-13 | 1998-04-29 |
| 1024 | 4444 | 1003 | 1996-03-11 | 1996-03-13 |
| 1025 | 5555 | 1004 | 1996-03-11 | 1996-03-12 |
| 1026 | 5555 | 1004 | 1998-05-31 | 1998-01-08 |
| 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 | 1985-09-17 |
| 1031 | 1000 | 1011 | 1993-05-26 | 1985-04-16 |
| 1032 | 1111 | 1012 | 1986-04-15 | 2005-01-31 |
| 1033 | 9988 | 1013 | 2000-04-18 | 2005-04-18 |
| 1034 | 9988 | 1014 | 2005-02-08 | 2005-02-18 |
| 1035 | 1111 | 1015 | 2005-04-18 | 2022-09-08 |
+------+------+------+------------+------------+
16 rows in set (0.00 sec)
```

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

Ans-

Query-

```
mysql>  update parts set price=(price-0.1*price) where price<24.00;
```

```
mysql> select * from parts;
+-------+--------------------+------+--------+--------+
| PNO   | PNAME              | QOH  | PRICE  | OLEVEL |
+-------+--------------------+------+--------+--------+
| 10506 | land before time 1 |  200 | 16.191 |     20 |
| 10507 | land before time 2 |  156 | 16.191 |     20 |
| 10508 | land before time 3 |  190 | 16.191 |     20 |
| 10509 | land before time 4 |   60 | 16.191 |     20 |
| 10601 | Nut                |  300 | 24.990 |     30 |
| 10602 | Bolt               |  120 | 16.191 |     30 |
| 10603 | Screw              |  140 | 12.141 |     30 |
| 10800 | Driller            |  100 | 24.990 |     30 |
+-------+--------------------+------+--------+--------+
8 rows in set (0.00 sec)
```

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

Ans-Query-

```
mysql> update parts set
   -> qoh=(
   -> select max(qoh) from
   -> (select qoh from parts) as t)
   -> where qoh<100;
```

```
mysql> select * from parts;
+-------+--------------------+------+--------+--------+
| PNO   | PNAME              | QOH  | PRICE  | OLEVEL |
+-------+--------------------+------+--------+--------+
| 10506 | land before time 1 |  200 | 17.990 |     20 |
| 10507 | land before time 2 |  156 | 17.990 |     20 |
| 10508 | land before time 3 |  190 | 17.990 |     20 |
| 10509 | land before time 4 |  300 | 17.990 |     20 |
| 10601 | Nut                |  300 | 24.990 |     30 |
| 10602 | Bolt               |  120 | 17.990 |     30 |
| 10603 | Screw              |  140 | 13.490 |     30 |
| 10800 | Driller            |  100 | 24.990 |     30 |
+-------+--------------------+------+--------+--------+
8 rows in set (0.00 sec)
```