ASSIGNMENT NO.4

Name – Indranil Bain
Enrollment NO. – 2020CSB039
Branch – Computer Science and Technology
Group - GX
Subject – DBMS Laboratory

Q1)
<A. Creation of tables:>

ZIPCODES Table
create table ZIPCODES(
   -> ZIP int primary key,
   -> CITY varchar(20));

```
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| ZIP   | int(11)     | NO   | PRI | NULL    |       |
| CITY  | varchar(20) | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
```

After insertion of values:-

```
+-------+-------------+
| 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     |
+-------+-------------+
```

EMPLOYEES Table
create table EMPLOYEES(
   -> ENO int primary key,
   -> ENAME varchar(20),
   -> ZIP int references ZIPCODES(ZIP),
   -> HDATE date);

```
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| ENO    | int(11)     | NO   | PRI | NULL    |       |
| ENAME  | varchar(20) | YES  |     | NULL    |       |
| ZIP    | int(11)     | YES  |     | NULL    |       |
| HDATE  | date        | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
```

After insertion of values:-

```
+------+----------+-------+------------+
| 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 | 1992-03-23 |
| 1014 | marium   | 61818 | 1993-03-22 |
| 1015 | falguni  | 61818 | 1988-03-11 |
+------+----------+-------+------------+
```

PARTS Table

create table PARTS(
        -> PNO int primary key,
        -> PNAME varchar(20),
        -> QOH int,
        -> PRICE double(4,2),
        -> OLEVEL int);

```
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| PNO    | int(11)     | NO   | PRI | NULL    |       |
| PNAME  | varchar(20) | YES  |     | NULL    |       |
| QOH    | int(11)     | YES  |     | NULL    |       |
| PRICE  | double(4,2) | YES  |     | NULL    |       |
| OLEVEL | int(11)     | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
```

insertion of values:-

```
+-------+----------------------+------+-------+--------+
| PNO   | PNAME                | QOH  | PRICE | OLEVEL |
+-------+----------------------+------+-------+--------+
| 10506 | land before time 1   |  200 | 17.99 |     20 |
| 10507 | land before time 2   |  156 | 17.99 |     20 |
| 10508 | land before time 3   |  190 | 17.99 |     20 |
| 10509 | land before time 4   |   60 | 17.99 |     20 |
| 10601 | Nut                  |  300 | 24.99 |     30 |
| 10602 | Bolt                 |  120 | 17.99 |     30 |
| 10603 | Screw                |  140 | 13.49 |     30 |
| 10800 | Driller              |  100 | 24.99 |     30 |
+-------+----------------------+------+-------+--------+
```

CUSTOMERS Table

create table CUSTOMERS(
    -> CNO int primary key,
    -> CNAME varchar(20),
    -> STREET varchar(20),
    -> ZIP int references ZIPCODES(ZIP),
    -> PHONE date);

```
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| CNO    | int(11)     | NO   | PRI | NULL    |       |
| CNAME  | varchar(20) | YES  |     | NULL    |       |
| STREET | varchar(20) | YES  |     | NULL    |       |
| ZIP    | int(11)     | YES  |     | NULL    |       |
| PHONE  | int(11)     | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
```

After insertion of values:-

```
+------+------+------+------------+------------+
| 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 | 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 | 1998-06-08 |
| 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 | 1985-09-17 |
| 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 | 2005-02-18 | NULL       |
+------+------+------+------------+------------+
```

ORDERS Table

create table ORDERS(
   -> ONO int primary key,
   -> CNO int references CUSTOMERS(CNO),
   -> ENO int references EMPLOYEES(ENO),
   -> RECEIVED date,
   -> SHIPPED date);

```
+----------+---------+------+-----+---------+-------+
| Field    | Type    | Null | Key | Default | Extra |
+----------+---------+------+-----+---------+-------+
| ONO      | int(11) | NO   | PRI | NULL    |       |
| CNO      | int(11) | YES  |     | NULL    |       |
| ENO      | int(11) | YES  |     | NULL    |       |
| RECEIVED | date    | YES  |     | NULL    |       |
| SHIPPED  | date    | YES  |     | NULL    |       |
+----------+---------+------+-----+---------+-------+
```

After insertion of values:-

```
+------+----------+--------------+-------+-----------+
| 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 |
+------+----------+--------------+-------+-----------+
```

ODETAILS Table

create table ODETAILS(
   -> ONO int references ORDERS(ONO),
   -> PNO int references PARTS(PNO),
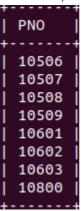   -> QTY int,
   -> primary key(ONO,PNO));

```
+-------+---------+------+-----+---------+-------+
| Field | Type    | Null | Key | Default | Extra |
+-------+---------+------+-----+---------+-------+
| ONO   | int(11) | NO   | PRI | NULL    |       |
| PNO   | int(11) | NO   | PRI | NULL    |       |
| QTY   | int(11) | YES  |     | NULL    |       |
+-------+---------+------+-----+---------+-------+
```

After insertion of values:-

```
+------+-------+------+
| ONO  | PNO   | QTY  |
+------+-------+------+
| 1020 | 10507 |    5 |
| 1021 | 10800 |    2 |
| 1022 | 10507 |    6 |
| 1026 | 10509 |    9 |
| 1026 | 10801 |    4 |
| 1028 | 10701 |    7 |
| 1029 | 10507 |    3 |
| 1029 | 10508 |    5 |
| 1031 | 10603 |    1 |
| 1032 | 10509 |    7 |
| 1033 | 10602 |    8 |
| 1034 | 10506 |    7 |
| 1035 | 10601 |    5 |
+------+-------+------+
```

<B. Queries >

1) Get PNO values for PARTS for which ORDERS have been placed.
SELECT DISTINCT(PNO) FROM ODETAILS;

PNO
10506
10507
10508
10509
10601
10602
10603
10800

2) Get all the details of CUSTOMERS whose name has the beginning letter 's'.
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

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

4) Get the ONO, CNAME and SHIPPED values for CUSTOMERS whose orders have not yet been shipped.
SELECT ONO, CNAME, SHIPPED FROM ORDERS,CUSTOMERS WHERE
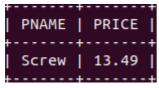ORDERS.CNO=CUSTOMERS.CNO AND SHIPPED IS NULL;

```
+------+---------+---------+
| ONO  | CNAME   | SHIPPED |
+------+---------+---------+
| 1022 | amina   | NULL    |
| 1035 | charles | NULL    |
+------+---------+---------+
```

5) Get PNAME and PRICE value from PARTS with the lowest PRICE.
SELECT PNAME, PRICE FROM PARTS WHERE PRICE = (SELECT MIN(PRICE) FROM PARTS);

```
+-------+-------+
| PNAME | PRICE |
+-------+-------+
| Screw | 13.49 |
+-------+-------+
```

6) Get the PNAME and PRICE values of PARTS that cost less than the least expensive 'land before time' part.
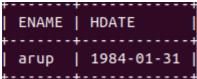SELECT PNAME, PRICE FROM PARTS WHERE PRICE < (SELECT MIN(PRICE) FROM PARTS WHERE PNAME LIKE 'LAND BEFORE TIME%');

```
+-------+-------+
| PNAME | PRICE |
+-------+-------+
| Screw | 13.49 |
+-------+-------+
```

7) Get the ENO values of EMPLOYEES from 'Fort Dodge'.
SELECT ENO FROM EMPLOYEES, ZIPCODES WHERE ZIPCODES.ZIP=EMPLOYEES.ZIP AND CITY='fort dodge';

```
+------+
| ENO  |
+------+
| 1001 |
+------+
```

8) Get the ENAME and HDATE of the EMPLOYEES who was hired on the earliest date.
SELECT ENAME, HDATE FROM EMPLOYEES WHERE HDATE = (SELECT MIN(HDATE) FROM EMPLOYEES);

```
+-------+------------+
| ENAME | HDATE      |
+-------+------------+
| arup  | 1984-01-31 |
+-------+------------+
```

9) Retrieve the PNO, PNAME and PRICE of PARTS with price greater than 20.00 in an ascending order of PNO.
SELECT PNO, PNAME, PRICE FROM PARTS WHERE PRICE > 20.00 ORDER BY PNO;

```
+-------+---------+-------+
| PNO   | PNAME   | PRICE |
+-------+---------+-------+
| 10601 | Nut     | 24.99 |
| 10800 | Driller | 24.99 |
+-------+---------+-------+
```

10) For each PARTS get PNO and PNAME values along with total sales in details.
SELECT PNO, PNAME, FORMAT(SUM(PRICE*QTY), 2) AS TOTAL_SALES FROM ODETAILS NATURAL JOIN PARTS GROUP BY PNO;
AFTER UPDATION

```
+--------+---------------------+--------------+
| PNO    | PNAME               | TOTAL_SALES  |
+--------+---------------------+--------------+
| 10506  | land before time 1  | 113.33       |
| 10507  | land before time 2  | 226.66       |
| 10508  | land before time 3  | 80.95        |
| 10509  | land before time 4  | 259.04       |
| 10601  | Nut                 | 124.95       |
| 10602  | Bolt                | 129.52       |
| 10603  | Screw               | 12.14        |
| 10800  | Driller             | 49.98        |
+--------+---------------------+--------------+
```

11) For each PARTS get PNO and PNAME values along with total sales in rupees but only when the total sales exceeds 100.53.
SELECT PNO, PNAME, FORMAT(SUM(price*QTY), 2) AS TOTAL_SALES FROM ODETAILS NATURAL JOIN PARTS GROUP BY PNO HAVING TOTAL_SALES > 100.53;
AFTER UPDATION

```
+--------+---------------------+--------------+
| PNO    | PNAME               | TOTAL_SALES  |
+--------+---------------------+--------------+
| 10506  | land before time 1  | 113.33       |
| 10507  | land before time 2  | 226.66       |
| 10509  | land before time 4  | 259.04       |
| 10601  | Nut                 | 124.95       |
| 10602  | Bolt                | 129.52       |
+--------+---------------------+--------------+
```

12) Change the name of the CITY 'columbia' to 'parth'.
UPDATE ZIPCODES
    -> SET CITY='PARTH'
    -> WHERE CITY='COLUMBIA';
AFTER UPDATION

```
+-------+-------------+
| 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     |
+-------+-------------+
```

13) Update all the null valued SHIPPED dates to the current date.
UPDATE ORDERS SET SHIPPED=(SELECT SYSDATE() FROM DUAL) WHERE
SHIPPED IS NULL;
AFTER UPDATION

```
+------+------+------+------------+------------+
| 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 | 2022-09-01 |
| 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 | 1998-06-08 |
| 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 | 1985-09-17 |
| 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 | 2005-02-18 | 2022-09-01 |
+------+------+------+------------+------------+
```

14) Decrease the PRICE of all PARTS that cost less than 24.00 by 10%.
UPDATE PARTS
   -> SET PRICE=PRICE-0.1*PRICE
   -> WHERE PRICE < 24.00;

AFTER UPDATION

```
+-------+---------------------+------+-------+--------+
| 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 |
+-------+---------------------+------+-------+--------+
```

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.
UPDATE PARTS
-> SET QOH = (SELECT MAX(QOH)
-> FROM (SELECT QOH FROM PARTS) AS T) -> WHERE QOH < 100;

AFTER UPDATION

```
+-------+---------------------+------+-------+--------+
| 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  | 300  | 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 |
+-------+---------------------+------+-------+--------+
```