Assignment – 16

Creating Tables and Indexes.

1) Write a command that will enable a user to pull orders grouped by date out of the Orders table quickly.

ANS :

KD4-89208-Ashish>create index id on orders(odate);

Query OK, 0 rows affected (0.05 sec)

Records: 0 Duplicates: 0 Warnings: 0

KD4-89208-Ashish>show indexes from orders;

```
KD4-89208-Ashish>drop index id on orders;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>create index id on orders(odate);
Query OK, 0 rows affected (0.05 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>show indexes from orders;
+--------+------------+----------+--------------+------------+-----------+-------------+
----------+---------+------------+
| Table  | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality |
x_comment | Visible | Expression |
+--------+------------+----------+--------------+------------+-----------+-------------+
----------+---------+------------+
| orders |          1 | id       |            1 | Odate       | A         |           3 |
          | YES     | NULL       |
+--------+------------+----------+--------------+------------+-----------+-------------+
----------+---------+------------+
1 row in set (0.04 sec)
```

KD4-89208-Ashish>select odate,count(*) from orders group by odate;

```
KD4-89208-Ashish>select odate,count(*) from orders group by odate;
+------------+----------+
| odate      | count(*) |
+------------+----------+
| 1990-10-03 |        5 |
| 1990-10-04 |        2 |
| 1990-10-06 |        1 |
+------------+----------+
3 rows in set (0.01 sec)
```

2) If the Orders table has already been created, how can you force the onum field to be unique (assume all current values are unique)?

ANS: KD4-89208-Ashish>alter table orders add unique index idx_onum(onum);

Query OK, 0 rows affected (0.05 sec)

Records: 0 Duplicates: 0 Warnings: 0

KD4-89208-Ashish>desc orders;

```
KD4-89208-Ashish>alter table orders add unique index idx_onum(onum);
Query OK, 0 rows affected (0.05 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>desc orders;
+-------+------------+------+-----+---------+-------+
| Field | Type       | Null | Key | Default | Extra |
+-------+------------+------+-----+---------+-------+
| Onum  | int        | YES  | UNI | NULL    |       |
| Amt   | float(7,2) | YES  |     | NULL    |       |
| Odate | date       | YES  |     | NULL    |       |
| Cnum  | int        | YES  |     | NULL    |       |
| Snum  | int        | YES  |     | NULL    |       |
+-------+------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

3) Create an index that would permit each salesperson to retrieve his or her orders grouped by date quickly.

ANS:

KD4-89208-Ashish>create index index_Snum_Odate on orders(snum,odate);

Query OK, 0 rows affected (0.04 sec)

Records: 0 Duplicates: 0 Warnings: 0

KD4-89208-Ashish>show indexes from orders;

```
KD4-89208-Ashish>create index index_Snum_Odate on orders(snum,odate);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>show indexes from orders;
+--------+------------+-----------------+--------------+-------------+-----------+-------------+----
| Table  | Non_unique | Key_name        | Seq_in_index | Column_name | Collation | Cardinality | Sub
+--------+------------+-----------------+--------------+-------------+-----------+-------------+----
| orders |          0 | idx_onum        |            1 | Onum        | A         |          10 |
| orders |          1 | id              |            1 | Odate       | A         |           4 |
| orders |          1 | index_Odate     |            1 | Odate       | A         |           4 |
| orders |          1 | index_Snum_Odate |           1 | Snum        | A         |           5 |
| orders |          1 | index_Snum_Odate |           2 | Odate       | A         |           9 |
+--------+------------+-----------------+--------------+-------------+-----------+-------------+----
5 rows in set (0.01 sec)
```

4) Let us assume that each salesperson is to have only one customer of a given rating, and that this is currently the case. Enter a command that enforces it.

ANS: KD4-89208-Ashish>drop index index_cnum_rating on customers;

Query OK, 0 rows affected (0.02 sec)

Records: 0 Duplicates: 0 Warnings: 0

KD4-89208-Ashish>create index index_cnum_rating on customers(cnum,rating);

Query OK, 0 rows affected (0.04 sec)

Records: 0 Duplicates: 0 Warnings: 0

KD4-89208-Ashish>select cnum,rating from customers group by cnum,rating;

```
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>create index index_cnum_rating on customers(cnum,rating);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

KD4-89208-Ashish>select cnum,rating from customers group by cnum,rating;
+------+--------+
| cnum | rating |
+------+--------+
| 2001 |    100 |
| 2002 |    300 |
| 2003 |    200 |
| 2004 |    300 |
| 2007 |    200 |
| 2008 |    300 |
+------+--------+
6 rows in set (0.01 sec)
```