**Experiment 1.1**

**Student Name:   Gaurav Kumar                                UID: 22MCC20177**

**Branch:   CC-DevOps                                                        Section/Group: 1/B**

**Semester:   I                                                               Date of Performance: 01/10/2022**

**Subject Name: ADBMS                     Subject Code: 22CAP-647**

1. **Task to be done:**

Given the table schemas below, write a query to print the company\_code, founder name, total number of lead managers, total number of senior managers, total number of managers, and total number of employees. Order your output by ascending company\_code.

**Steps for experiment/practical:**

**mysql>** create database MCA;

**mysql>** use MCA;

**mysql>** create table Company (

-> companyCode varchar(4) primary key,

-> founder varchar(30)

-> );

**Query OK, 0 rows affected (0.02 sec)**

**mysql>** create table LeadManager (

-> leadManagerCode varchar(5) primary key,

-> companyCode varchar(4),

-> FOREIGN KEY(companyCode) REFERENCES Company(companyCode)

-> );

**Query OK, 0 rows affected (0.04 sec)**

**mysql>** create table SeniorManager (

-> seniorManagerCode varchar(5) primary key,

-> leadManagerCode varchar(5),

-> companyCode varchar(4),

-> FOREIGN KEY(leadManagerCode) REFERENCES LeadManager(leadManagerCode),

-> FOREIGN KEY(companyCode) REFERENCES Company(companyCode)

-> );

**Query OK, 0 rows affected (0.03 sec)**

**mysql>** create table Manager (

-> ManagerCode varchar(5) primary key,

-> seniorManagerCode varchar(5),

-> leadManagerCode varchar(5),

-> companyCode varchar(4),

-> FOREIGN KEY(seniorManagerCode) REFERENCES SeniorManager(seniorManagerCode),

-> FOREIGN KEY(leadManagerCode) REFERENCES LeadManager(leadManagerCode),

-> FOREIGN KEY(companyCode) REFERENCES Company(companyCode)

-> );

**Query OK, 0 rows affected (0.11 sec)**

**mysql>** create table Employee (

-> employeeCode varchar(4) primary key,

-> ManagerCode varchar(5),

-> seniorManagerCode varchar(5),

-> leadManagerCode varchar(5),

-> companyCode varchar(4),

-> FOREIGN KEY(ManagerCode) REFERENCES Manager(ManagerCode),

-> FOREIGN KEY(seniorManagerCode) REFERENCES SeniorManager(seniorManagerCode),

-> FOREIGN KEY(leadManagerCode) REFERENCES LeadManager(leadManagerCode),

-> FOREIGN KEY(companyCode) REFERENCES Company(companyCode)

-> );

**Query OK, 0 rows affected (0.04 sec)**

**mysql>** insert into company values("C1","Monika"),

-> ("C2","Samantha");

**Query OK, 2 rows affected (0.01 sec)**

**Records: 2 Duplicates: 0 Warnings: 0**

**mysql>** insert into leadManager values("LM1","C1"),

-> ("LM2","C2");

**Query OK, 2 rows affected (0.01 sec)**

**Records: 2 Duplicates: 0 Warnings: 0**

**mysql>** insert into seniorManager values("SM1","LM1","C1"),

-> ("SM2","LM1","C1"),

-> ("SM3","LM2","C2");

**Query OK, 3 rows affected (0.01 sec)**

**Records: 3 Duplicates: 0 Warnings: 0**

**mysql>** insert into Manager values("M1","SM1","LM1","C1"),

-> ("M2","SM3","LM2","C2"),

-> ("M3","SM3","LM2","C2");

**Query OK, 3 rows affected (0.01 sec)**

**Records: 3 Duplicates: 0 Warnings: 0**

**mysql>** insert into Employee values("E1","M1","SM1","LM1","C1"),

-> ("E2","M1","SM1","LM1","C1"),

-> ("E3","M2","SM3","LM2","C2"),

-> ("E4","M3","SM3","LM2","C2");

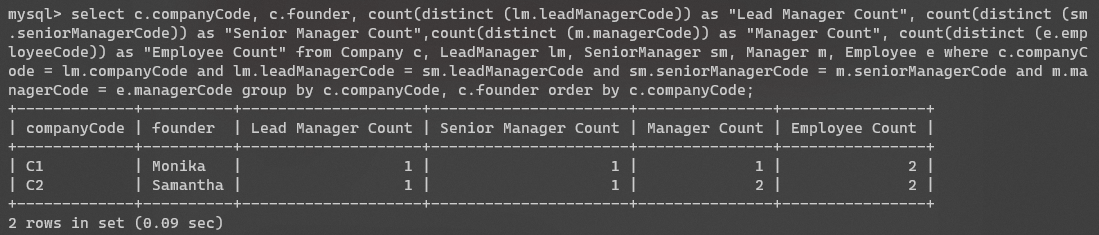
**Query OK, 4 rows affected (0.01 sec)**

**Records: 4 Duplicates: 0 Warnings: 0**

**Query :**

**select c.companyCode, c.founder, count(distinct (lm.leadManagerCode)) as "Lead Manager Count", count(distinct (sm.seniorManagerCode)) as "Senior Manager Count",count(distinct (m.managerCode)) as "Manager Count", count(distinct (e.employeeCode)) as "Employee Count" from Company c, LeadManager lm, SeniorManager sm, Manager m, Employee e where c.companyCode = lm.companyCode and lm.leadManagerCode = sm.leadManagerCode and sm.seniorManagerCode = m.seniorManagerCode and m.managerCode = e.managerCode group by c.companyCode, c.founder order by c.companyCode;**

1. **Output (screenshots)**

****

1. **Learning outcomes (What I have learnt):** 
   * 1. **Use of as Keyword to rename column name**
     2. **Order by**
     3. **Group by**

**Evaluation Grid:**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. | Demonstration and Performance  (Quiz) |  | 22 |
| 2. | Worksheet |  | 8 |