**INTRODUCTION TO DATABASE SYSTEM LAB**

**PROJECT NO 01**

**PROJECT PHASE 01**

**NAME:** MUHAMMAD FAROOQ AZAM

**REG #:** L1F20BSSE0593

**SECTION:** O1

**PROJECT TITLE**

**PHARMACY**

**Database Name:** pharmacy

**Table: Company**

* Name (Primary Key)
* Address
* PhoneNumber

**Table: Admin**

* Admin\_id (Primary Key)
* Name
* Address
* PhoneNumber

**Table: Pharmacist**

* Pharm\_id (Primary Key)
* Name
* Address
* PhoneNumber

**Table: Customer**

* Cust\_id (Primary Key)
* Name
* Address
* PhoneNumber

**Table: Supplier**

* Supp\_id (Primary Key)
* Name
* Address
* PhoneNumber
* Comp\_name (Foreign Key)

**Table: Medicines**

* Med\_id (Primary Key)
* Name
* Comp\_name (Foreign Key)
* Supp\_id (Foreign Key)
* Price
* ExpiryDate

**Table: Prescription**

* Presc\_id (Primary Key)
* Med\_id (Foreign Key)
* Hospitalname
* Doctorname
* PreDate

**Table: Purchase**

* Purchase\_id (Primary Key)
* Cust\_id (Foreign Key)
* Med\_id (Foreign Key)
* Amount
* PurchaseDate

**Table: Sales**

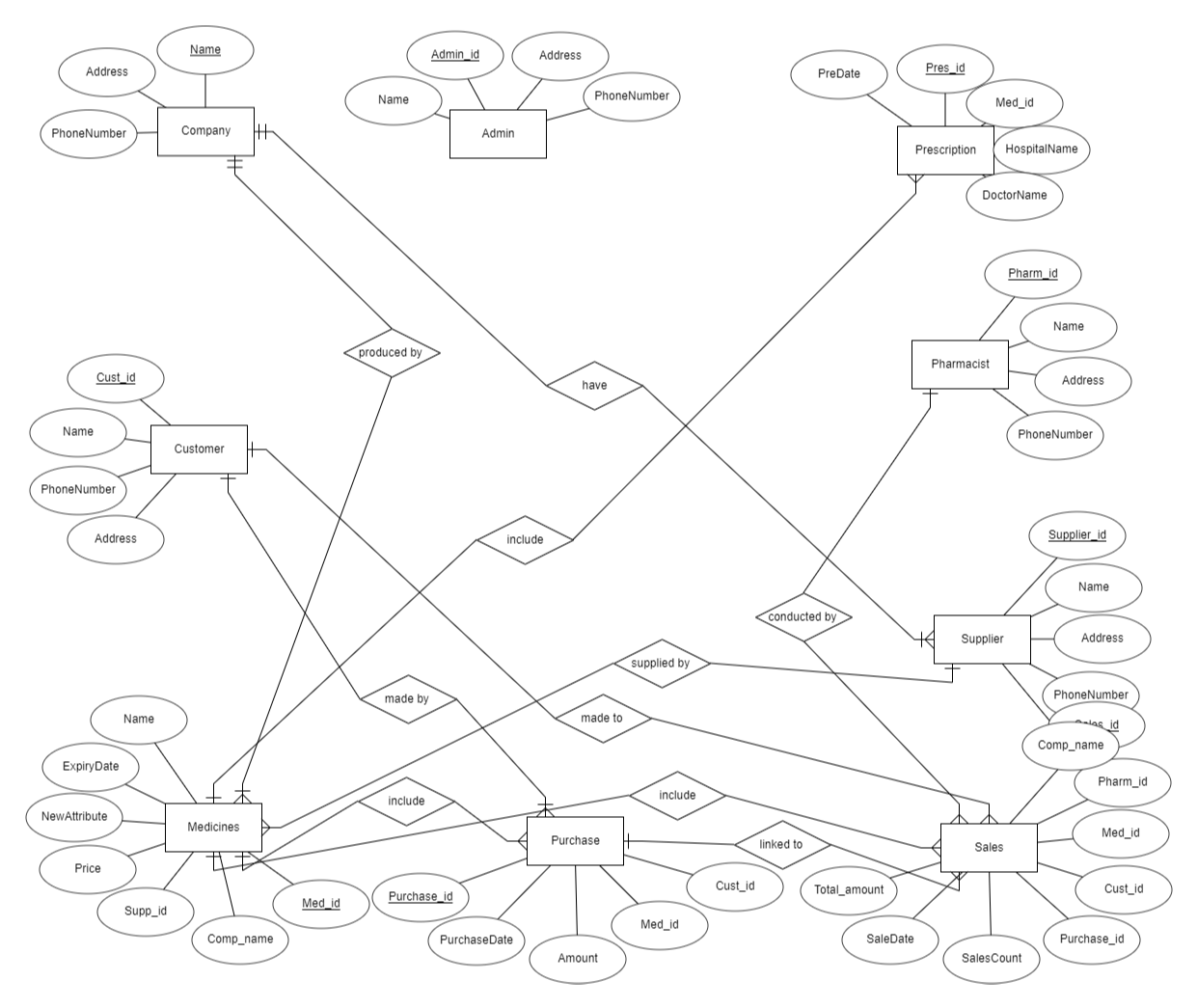
* Sales\_id (Primary Key)
* Pharm\_id (Foreign Key)
* Med\_id (Foreign Key)
* Cust\_id (Foreign Key)
* Purchase\_id (Foreign Key)
* SalesCount
* SaleDate
* Total\_amount

**Relationships:**

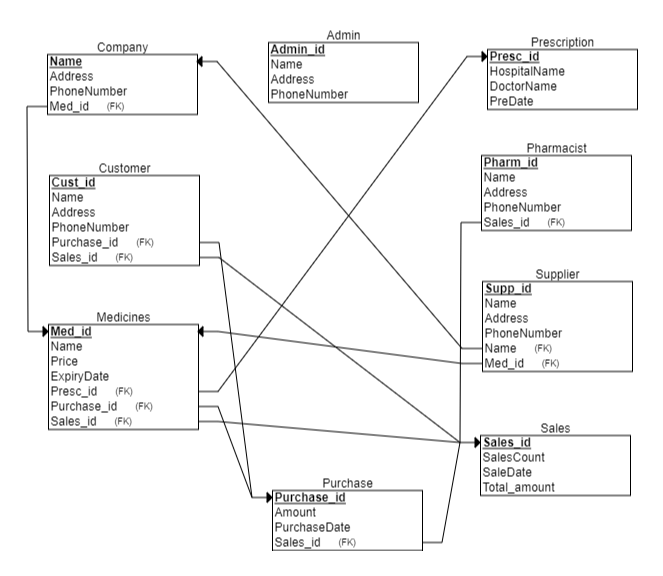
1. **Company - Supplier (One-to-Many):**
   * Each company (Company) can have one or more suppliers (Supplier).
   * The Supplier table contains a foreign key (Comp\_name) referencing the Company table.
2. **Medicines - Company (Many-to-One):**
   * Many medicines (Medicines) are produced by one company (Company).
   * The Medicines table contains a foreign key (Comp\_name) referencing the Company table.
3. **Medicines - Supplier (Many-to-One):**
   * Many medicines (Medicines) can be supplied by one supplier (Supplier).
   * The Medicines table contains a foreign key (Supp\_id) referencing the Supplier table.
4. **Prescription - Medicines (Many-to-One):**
   * Many prescriptions (Prescription) can include one type of medicine (Medicines).
   * The Prescription table contains a foreign key (Med\_id) referencing the Medicines table.
5. **Purchase - Customer (Many-to-One):**
   * Many purchases (Purchase) can be made by one customer (Customer).
   * The Purchase table contains a foreign key (Cust\_id) referencing the Customer table.
6. **Purchase - Medicines (Many-to-One):**
   * Many purchases (Purchase) can include one type of medicine (Medicines).
   * The Purchase table contains a foreign key (Med\_id) referencing the Medicines table.
7. **Sales - Pharmacist (Many-to-One):**
   * Many sales (Sales) can be conducted by one pharmacist (Pharmacist).
   * The Sales table contains a foreign key (Pharm\_id) referencing the Pharmacist table.
8. **Sales - Medicines (Many-to-One):**
   * Many sales (Sales) can include one type of medicine (Medicines).
   * The Sales table contains a foreign key (Med\_id) referencing the Medicines table.
9. **Sales - Customer (Many-to-One):**
   * Many sales (Sales) can be made to one customer (Customer).
   * The Sales table contains a foreign key (Cust\_id) referencing the Customer table.
10. **Sales - Purchase (Many-to-One):**
    * Many sales (Sales) can be linked to one purchase (Purchase).
    * The Sales table contains a foreign key (Purchase\_id) referencing the Purchase table.

**PROJECT PHASE 02**

**ER MODEL**



**RELATIONAL MODEL**



**PROJECT PHASE 03**

**Queries**

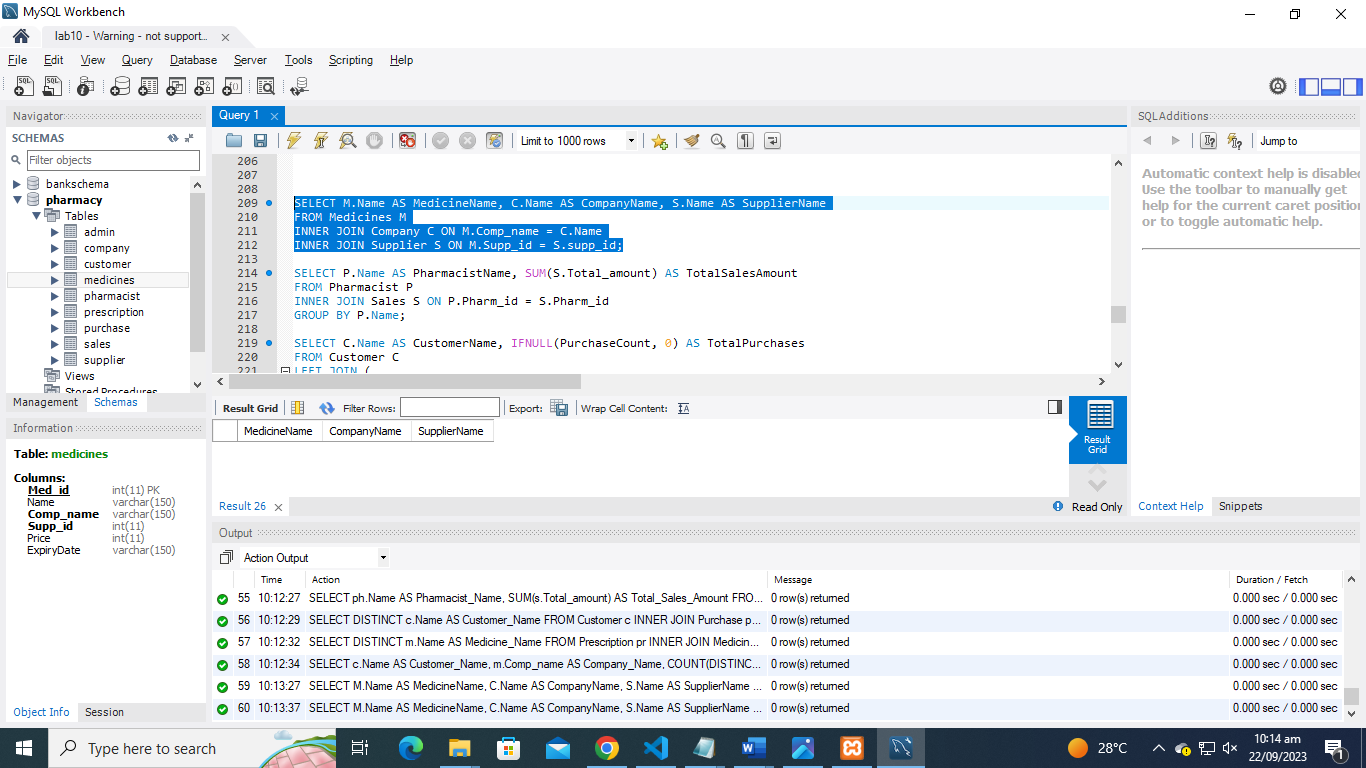
**Query 01:**

SELECT M.Name AS MedicineName, C.Name AS CompanyName, S.Name AS SupplierName

FROM Medicines M

INNER JOIN Company C ON M.Comp\_name = C.Name

INNER JOIN Supplier S ON M.Supp\_id = S.supp\_id;



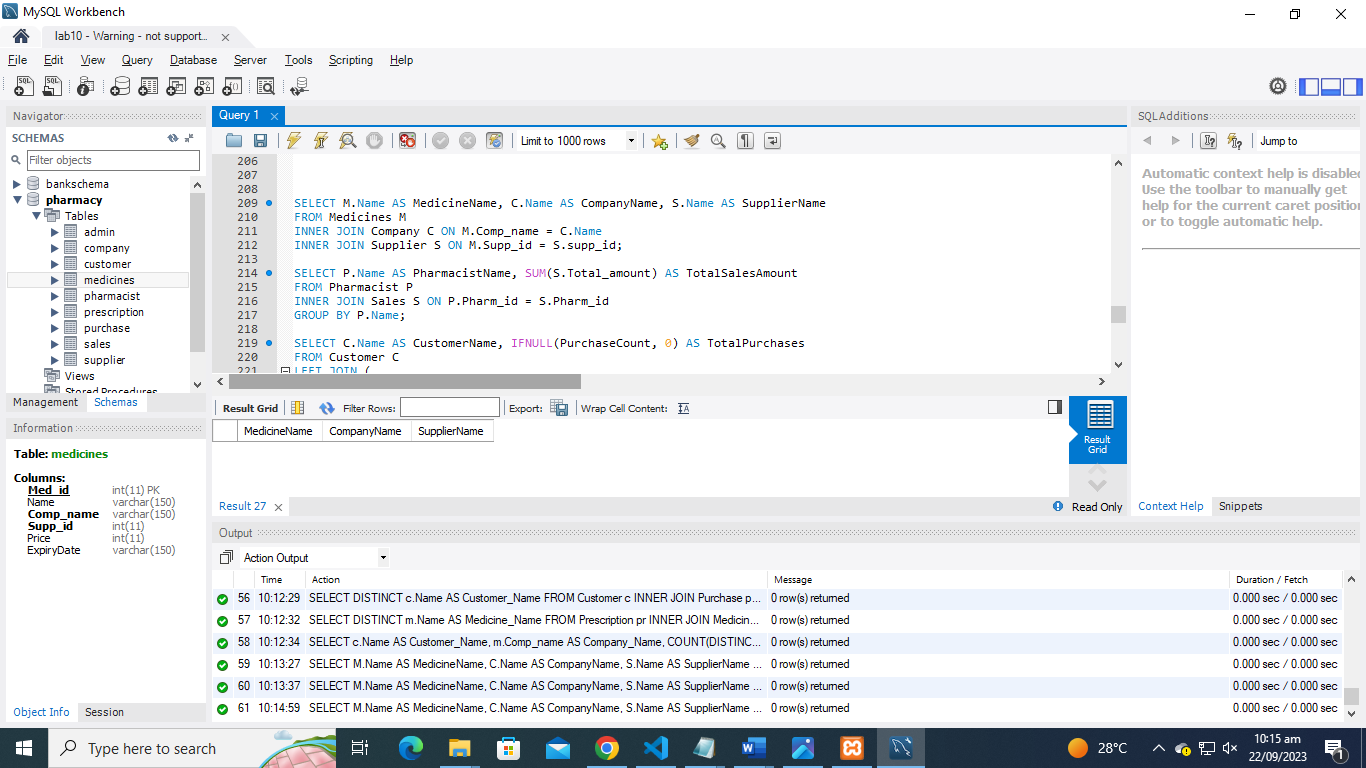
**Query 02:**

SELECT P.Name AS PharmacistName, SUM(S.Total\_amount) AS TotalSalesAmount

FROM Pharmacist P

INNER JOIN Sales S ON P.Pharm\_id = S.Pharm\_id

GROUP BY P.Name;



**Query 03:**

SELECT C.Name AS CustomerName, IFNULL(PurchaseCount, 0) AS TotalPurchases

FROM Customer C

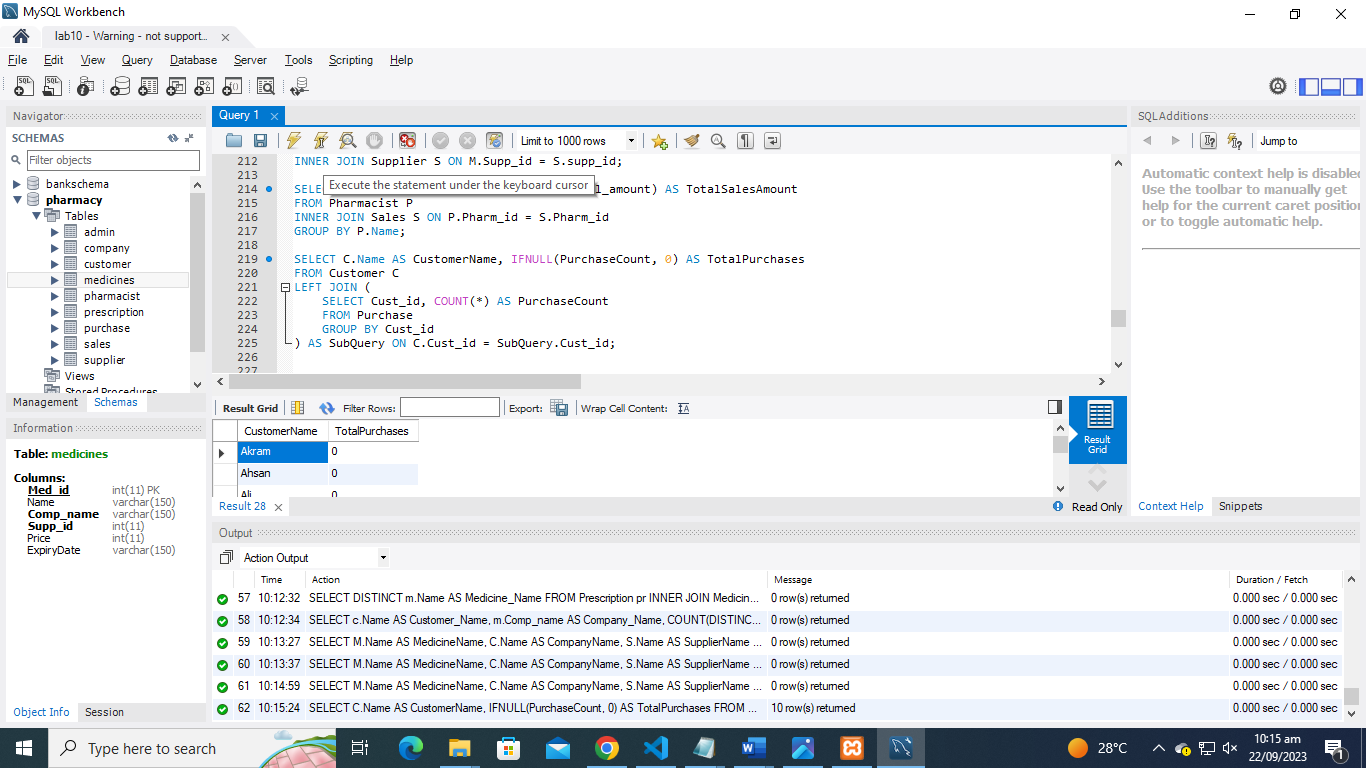
LEFT JOIN (

SELECT Cust\_id, COUNT(\*) AS PurchaseCount

FROM Purchase

GROUP BY Cust\_id

) AS SubQuery ON C.Cust\_id = SubQuery.Cust\_id;



**Query 04:**

SELECT M.Name AS MedicineName, IFNULL(PrescriptionCount, 0) AS TotalPrescriptions

FROM Medicines M

LEFT JOIN (

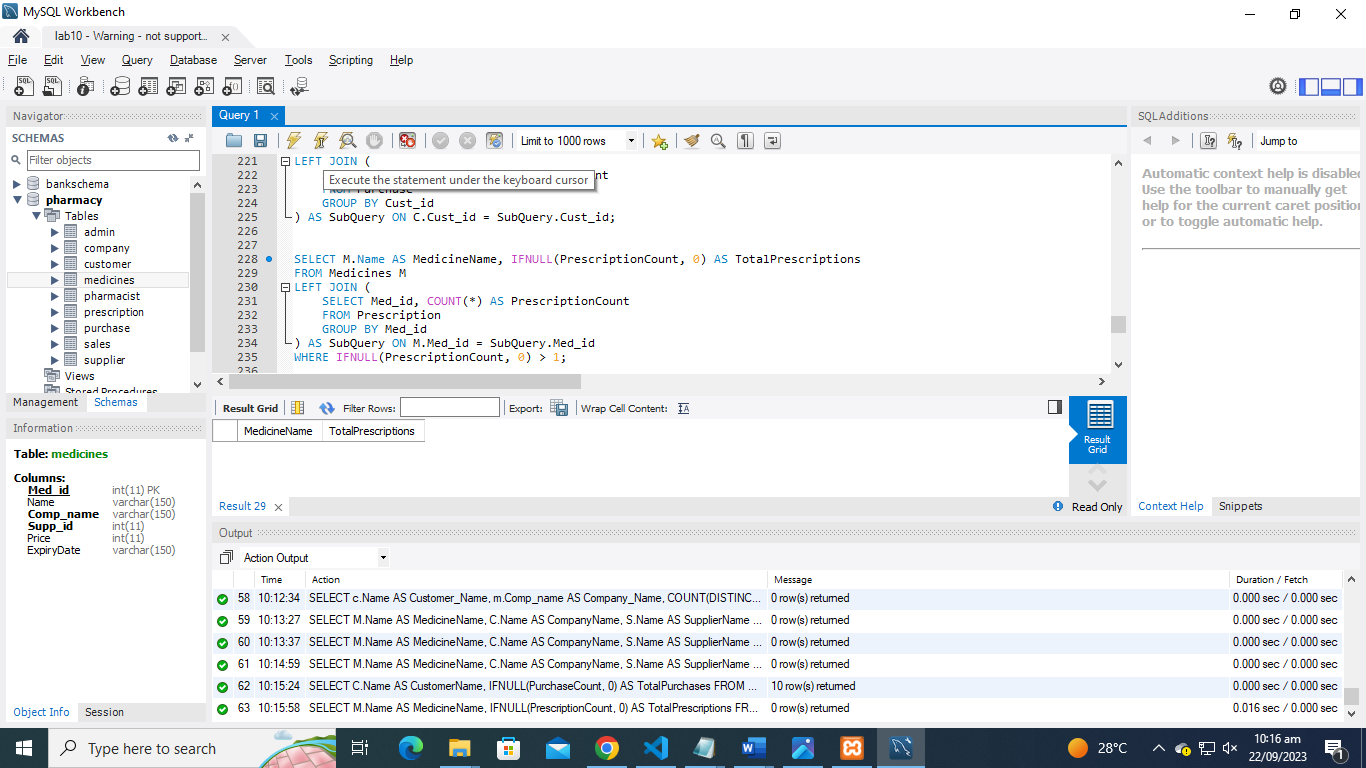
SELECT Med\_id, COUNT(\*) AS PrescriptionCount

FROM Prescription

GROUP BY Med\_id

) AS SubQuery ON M.Med\_id = SubQuery.Med\_id

WHERE IFNULL(PrescriptionCount, 0) > 1;



**Query 05:**

SELECT C.Name AS CustomerName, (

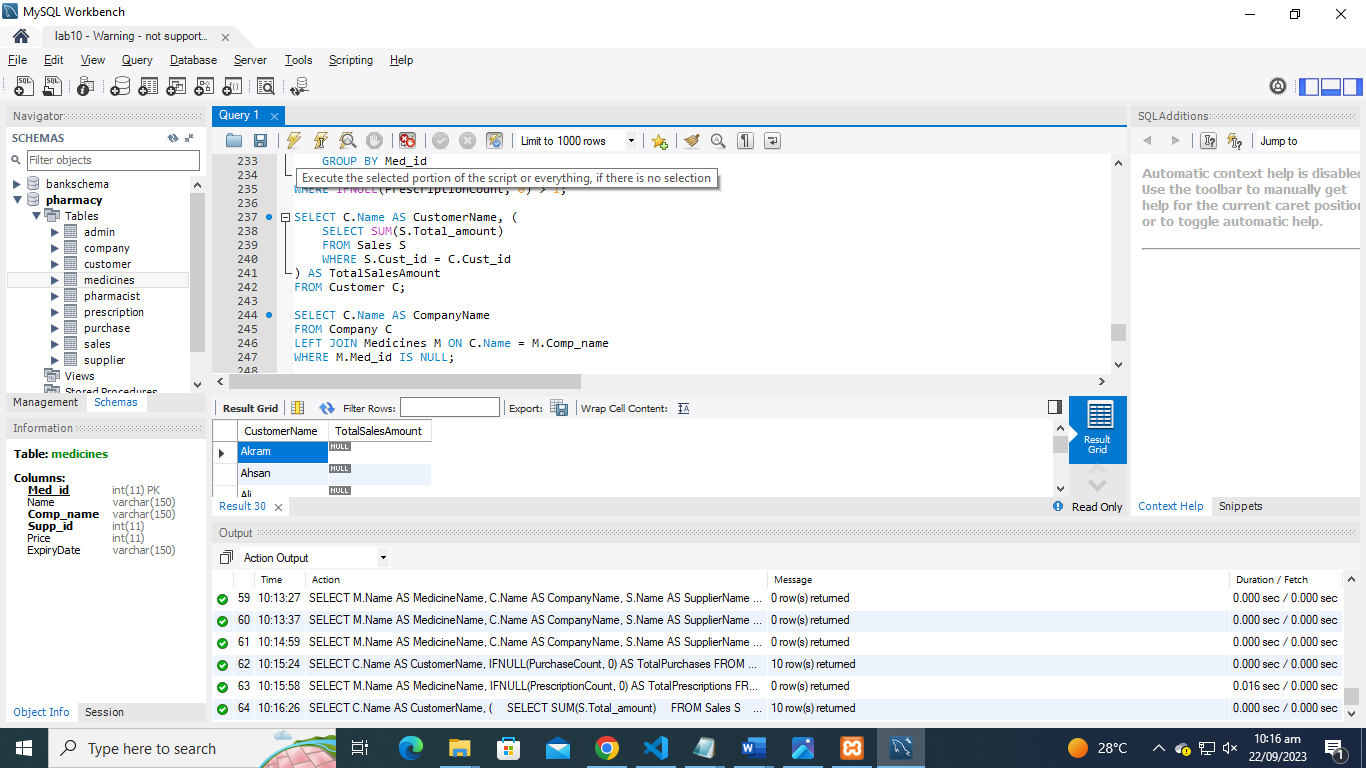
SELECT SUM(S.Total\_amount)

FROM Sales S

WHERE S.Cust\_id = C.Cust\_id

) AS TotalSalesAmount

FROM Customer C;



**Query 06:**

SELECT C.Name AS CompanyName

FROM Company C

LEFT JOIN Medicines M ON C.Name = M.Comp\_name

WHERE M.Med\_id IS NULL;

