Material Procurement Management: A Database Design Case Study

Introduction

Material procurement is a vital process in the construction industry, responsible for acquiring necessary materials and ensuring that they are available when required. Managing procurement effectively involves tracking suppliers, managing inventory, and ensuring timely deliveries to avoid project delays. This case study demonstrates the development of a Material Procurement Management Database using database design principles to create an efficient system.

Mission

The mission of this database system is to streamline the procurement process by creating an efficient and scalable structure for managing materials, suppliers, inventory, and deliveries. The goal is to optimize resource allocation and ensure accurate and timely data to support decision-making in construction projects.

Objective

The primary objectives of this case study are:
1. To develop a comprehensive database system for managing procurement operations.
2. To ensure that the database structure is scalable and can handle the complexity of multiple projects, suppliers, and deliveries.
3. To provide accurate reports and real-time tracking of inventory, orders, and deliveries.
4. To ensure efficient data management through the use of relational database models.

Database Design

The design of the Material Procurement Management Database includes identifying key entities such as suppliers, materials, clients, employees, and projects, and defining the relationships between these entities. This allows the system to track procurement activities from supplier to project completion.

Why Database Design is Important

A well-structured database ensures that the material procurement process is smooth and efficient, reducing the risk of errors, delays, and resource misallocation. It also helps manage complex relationships between multiple suppliers, materials, and construction

projects. By creating a clear structure, the database improves data accuracy, supports reporting, and enhances decision-making.
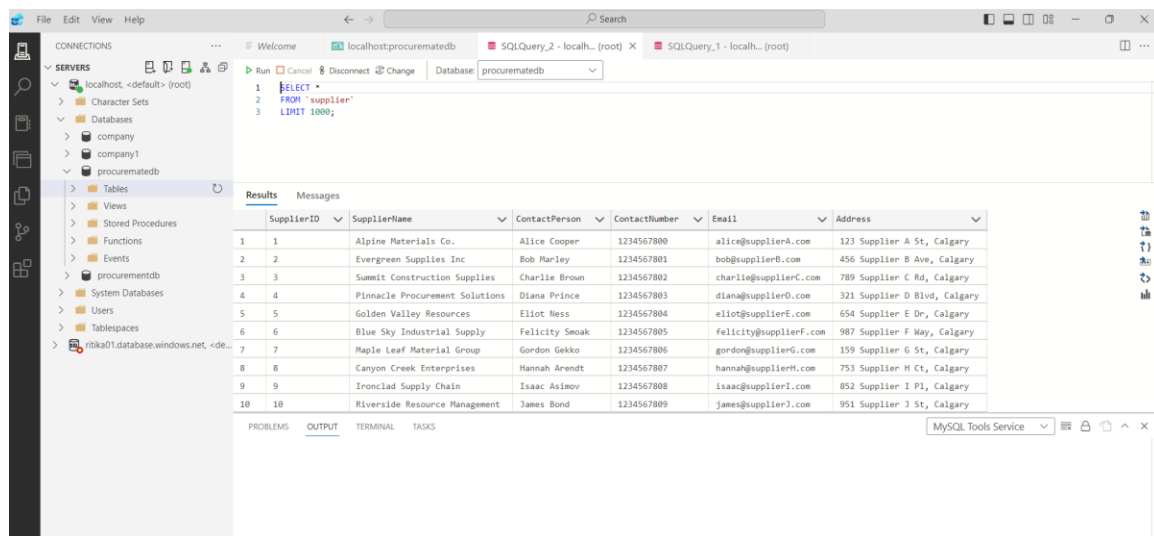
Database Structure: Key Entities and Their Tables

Supplier Table

This table holds information about the suppliers who provide materials for construction projects.

Attributes:
1. SupplierID (Primary Key)
2. SupplierName
3. ContactPerson
4. ContactNumber
5. Email (Unique)
6. Address

Material Table

The Material table stores details of the various materials used in construction.

Attributes:
1. MaterialID (Primary Key)
2. MaterialName
3. UnitPrice
4. UnitOfMeasure
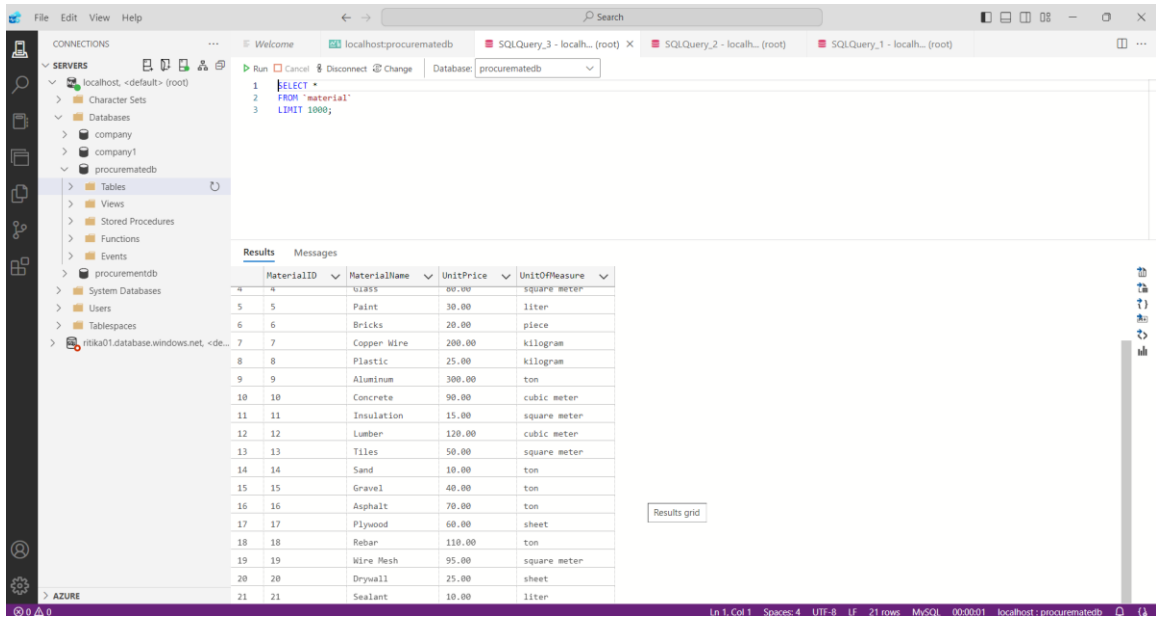
Client Table

Stores information about clients commissioning construction projects.

Attributes:
1. ClientID (Primary Key)
2. ClientName
3. ContactPerson
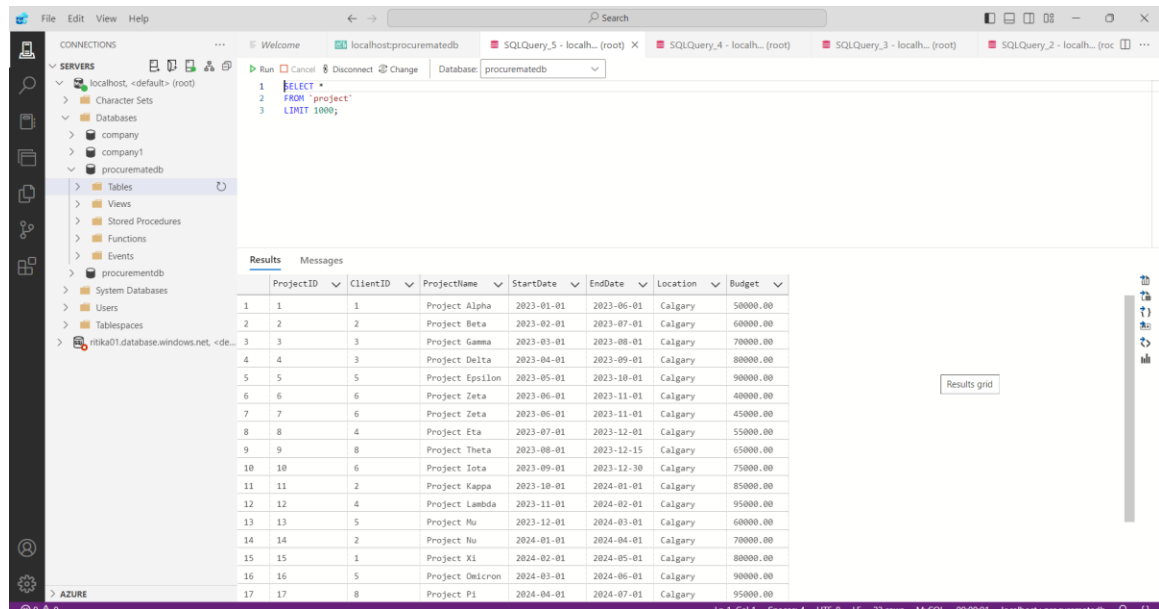4. ContactNumber
5. Email
6. Address
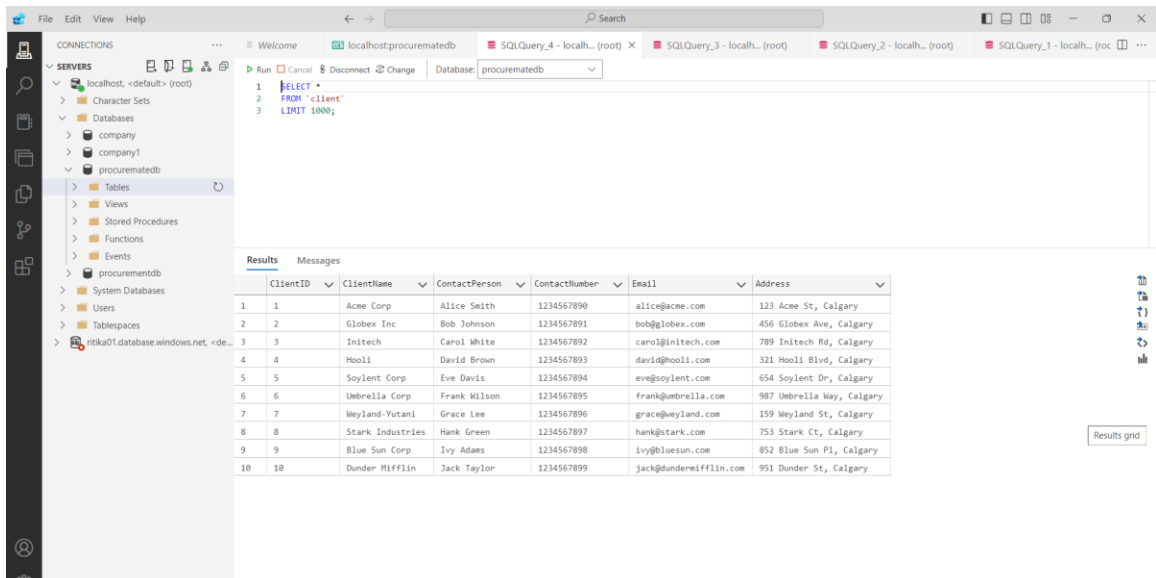
Project Table

Captures all details related to individual construction projects.

Attributes:
1. ProjectID (Primary Key)
2. ClientID (Foreign Key referencing Client)
3. ProjectName
4. StartDate
5. EndDate
6. Budget

Delivery Table

Tracks the delivery details for orders made to suppliers.

Attributes:
1. DeliveryID (Primary Key)
2. OrderID (Foreign Key referencing Order)
3. EmployeeID (Foreign Key referencing Employee)
4. DeliveryDate
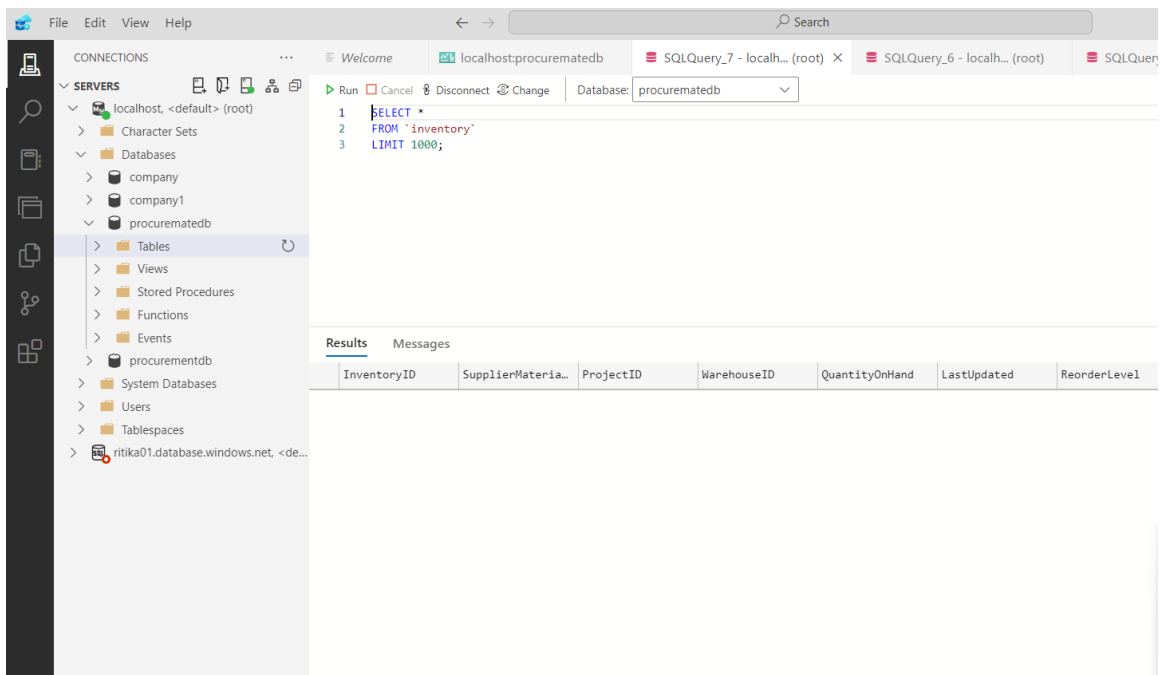5. QuantityDelivered
6. DeliveryNote
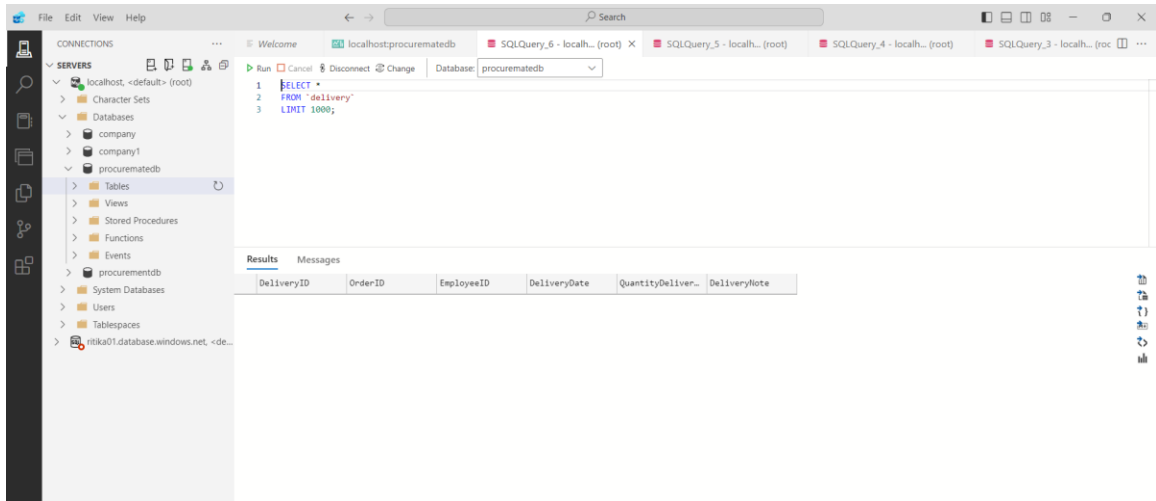
Inventory Table

Contains inventory information across different warehouse locations.

Attributes:
1. InventoryID (Primary Key)
2. SupplierMaterialID (Foreign Key referencing SupplierMaterial)
3. WarehouseID (Foreign Key referencing Warehouse)
4. QuantityOnHand
5. LastUpdated
6. ReorderLevel

Warehouse Table

Stores details about the warehouses that store materials.

Attributes:
1. WarehouseID (Primary Key)
2. WarehouseName
3. Location
4. Capacity

Employee Table

Holds information about employees responsible for various tasks.

Attributes:
1. EmployeeID (Primary Key)
2. FirstName
3. LastName
4. Position
5. ContactNo
6. Email
7. DepartmentID (Foreign Key referencing Department)

Department Table

Stores information about the different departments employees belong to.

Attributes:
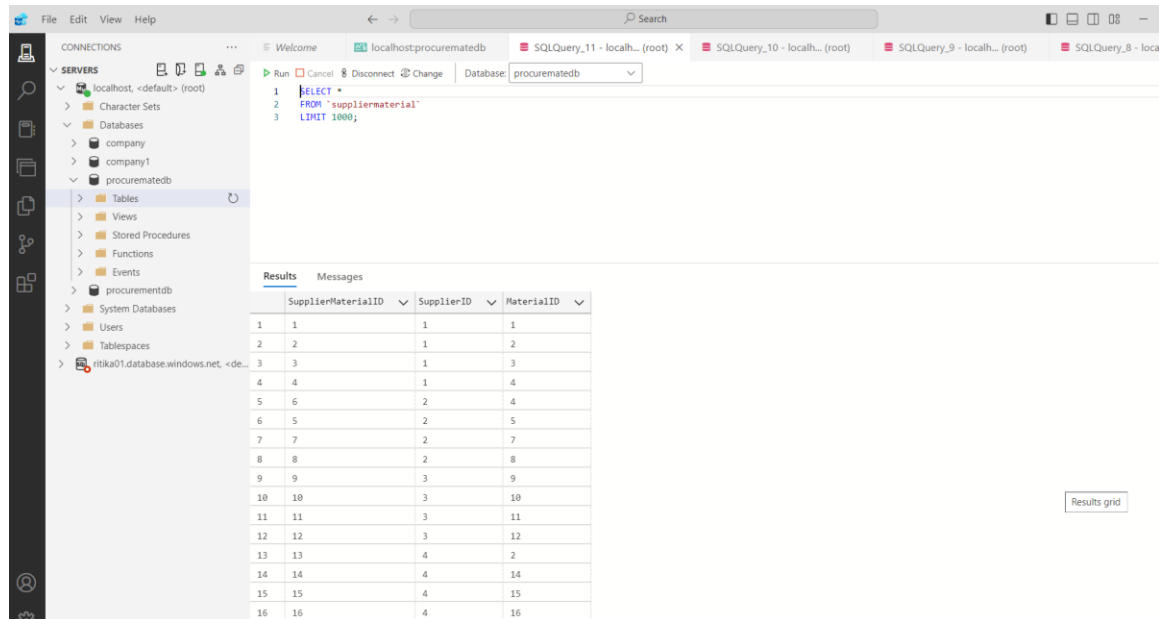1. DepartmentID (Primary Key)
2. DepartmentName

SupplierMaterial Table

Contains information about materials supplied by different suppliers.

Attributes:
1. SupplierMaterialID (Primary Key)
2. SupplierID (Foreign Key referencing Supplier)
3. MaterialID (Foreign Key referencing Material)

Entity Relationships

The database design includes various relationships that help in managing the procurement process efficiently:

1. Client to Project: One client can have multiple projects (One-to-Many).
2. Supplier to Material: One supplier can provide multiple materials, and one material can come from multiple suppliers (Many-to-Many).
3. Warehouse to Inventory: A warehouse can store multiple inventories (One-to-Many).
4. Project to Order: Each project can generate multiple orders (One-to-Many).

ER Diagram

The ER diagram visually represents how the entities (tables) are related to one another, ensuring smooth data flow between entities.
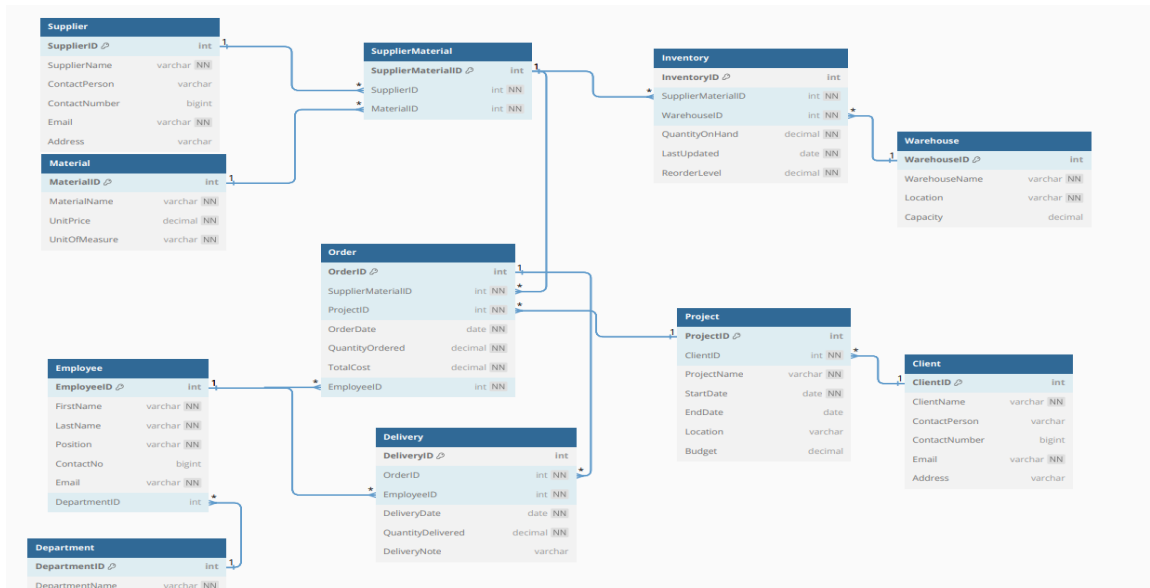
Why This Structure?

The chosen database structure supports the efficient handling of complex relationships and data within the procurement process. This design ensures that:

1. Data integrity is maintained through primary and foreign keys.
2. Complex relationships are represented using relational models.
3. Scalability is achieved by allowing new entities or relationships to be easily added.

Conclusion

The Material Procurement Management System is essential for any construction company aiming to streamline its procurement process. A well-designed database ensures data accuracy, reduces delays, and improves resource allocation, ultimately contributing to the success of construction projects.

Appendix A - Table Details

1. Supplier Table: Detailed information about suppliers.
2. Client Table: Information on clients commissioning projects.
3. Material Table: Detailed material information.
4. Project Table: All relevant project data.
5. Employee Table: Details of employees handling procurement.
6. Order Table: Tracks procurement orders.

Appendix B - Test Queries (Sample MySQL)

1. Query to Find Suppliers for a Specific Material:
   ```sql

```sql
SELECT SupplierName
FROM Supplier
JOIN SupplyMaterial ON Supplier.SupplierID = SupplyMaterial.SupplierID
WHERE MaterialID = (SELECT MaterialID FROM Material WHERE MaterialName =
'Steel');
```

2. Query to Track Orders for a Specific Project:
```sql
SELECT OrderID, ProjectID, QuantityOrdered, TotalCost
FROM Orders
WHERE ProjectID = 22;
```

3. Query to Track Deliveries for a Specific Order:
```sql
SELECT DeliveryID, OrderID, QuantityDelivered, DeliveryDate
FROM Delivery
WHERE OrderID = 31;
```
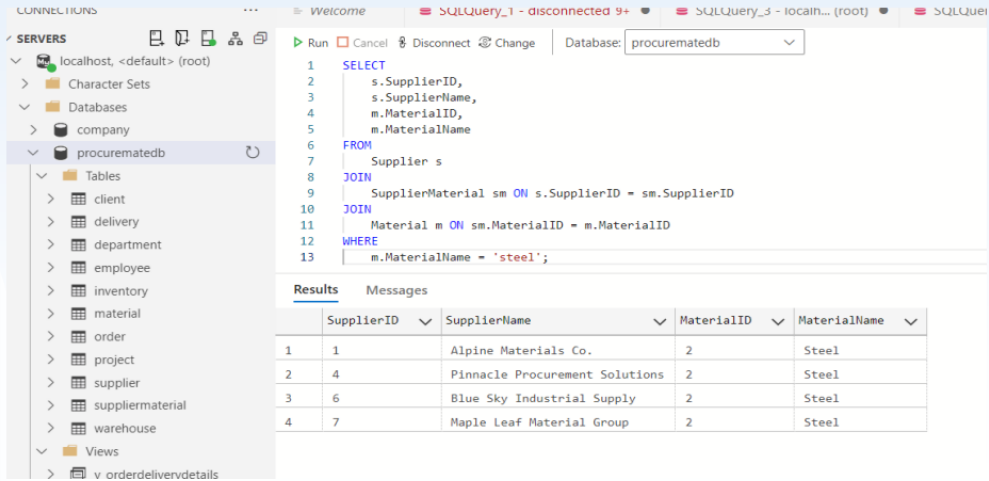
Screenshots of Join Queries

Query 1: Identify Suppliers for Specific Material (Steel)

This query identifies suppliers who provide steel material. The SQL code and its result are shown below.

Query 2: View Inventory Status

This query shows the inventory status of materials across warehouse locations.

Query 2 /view

provides valuable insights into the inventory status of various materials across different warehouse locations.(facts from Inventory table)

Query 3: Materials Ordered for a Specific Project

This query retrieves all materials ordered for a specific project (e.g., Project ID 22).

Query 3:
materials ordered for a specific project(ex projectid:22)

Query 4: Track Deliveries for a Specific Order

This query tracks all deliveries made for a specific order (e.g., Order ID 31).