Table of Contents

List of Figures

1 Chapter 1-Requirement Analysis

Introduction

In today's rapidly advancing world, efficient waste management systems have become imperative for sustainable development and environmental preservation. Waste management not only ensures the cleanliness of our surroundings but also plays a crucial role in resource conservation and pollution reduction. As we strive towards creating a greener and healthier planet, the need for effective waste management practices is more pronounced than ever.

This mini-project endeavors to address the complexities inherent in managing and optimizing waste management systems. With waste generation on the rise due to population growth and urbanization, it is crucial to develop robust strategies for waste collection, disposal, and recycling. The database system proposed here aims to streamline these processes, offering a comprehensive solution for managing waste-related operations in Sri Lanka.

By delving into the organizational and operational intricacies of waste management departments, this project seeks to design and implement a database management system tailored to the specific needs of waste management authorities in Sri Lanka. Through careful analysis of functional and data requirements, the system aims to enhance the efficiency and effectiveness of waste management processes, from waste collection and segregation to recycling and disposal.

This project aims to develop a sophisticated database management system tailored to the needs of waste management in Sri Lanka. By providing a user-friendly interface and robust functionality, the system empowers administrative staff and stakeholders to make informed decisions, optimize resource allocation, and improve overall waste management outcomes. Through the utilization of data and technology, this project seeks to address the challenges associated with waste management, ultimately contributing to a cleaner, healthier, and more sustainable future for generations to come.

Functional Requirements

1. User authentication and authorization: The system should allow authorized users, such as administrative staff and waste management personnel, to log in securely with appropriate access levels.

2. Waste collection management: The system should facilitate the scheduling, tracking, and management of waste collection activities, including routes, pickup times, and types of waste to be collected.

3. Waste segregation and classification : The system should support the classification and segregation of different types of waste, such as recyclables, organic waste, and hazardous materials, to ensure proper handling and disposal.

4. Inventory Management: The system should maintain an inventory of waste management resources, including bins, containers, vehicles, and equipment, and track their availability, usage, and maintenance schedules.

5. Customer Management: The system should enable the registration and management of customers, such as households, businesses, and industrial facilities, including contact information, service subscriptions, and billing details.

6. Compliance and regulatory requirements: The system should help ensure compliance with local regulations and environmental standards governing waste management practices, including documentation and reporting obligations.

Data Requirements

Data requirements specify the specific data elements and attributes that the waste management database system needs to manage. Here are the data requirements for the waste management project.

List of the entities

1. User
2. WasteBin
3. WasteType
4. WasteCollector
5. WasteCollection
6. RegulatoryAgency
7. ViolationRecord
8. LandFill
9. WasteItem
10. ProcessingFacility
11. ProceedWaste

Entity	Attributes
User	Name
	Address
	Contact_Number
	User_ID
	User_Type
	User_Status
WasteBin	Bin_ID
	User_ID
	Location
	Capacity
	Bin_type
	Current_Fill_Level
WasteType	Type_ID
	Type_Name
WasteCollector	Name
	Collector_ID
	Contact_Number
WasteCollection	Collector_ID
	Bin_ID
	Collection_ID
	Collection_Date
RegulatoryAgency	Agency_ID
	Name
	Contact_Information
ViolationRecord	Agency_ID
	Description
	Date
LandFill	LandFill_ID
	Agency_ID
	Type_ID
	Capacity
	Location
WasteItem	Type_ID
	Collection_ID
	LandFill_ID
	Date
ProcessingFacility	Facility_ID
	Facility_Type
	Location
	Type_ID
	Proceed_ID

ProceedWaste	Facility_ID
	Data_Proceed
	Quantity

2 Chapter 2-Conceptual Design

ER Diagram

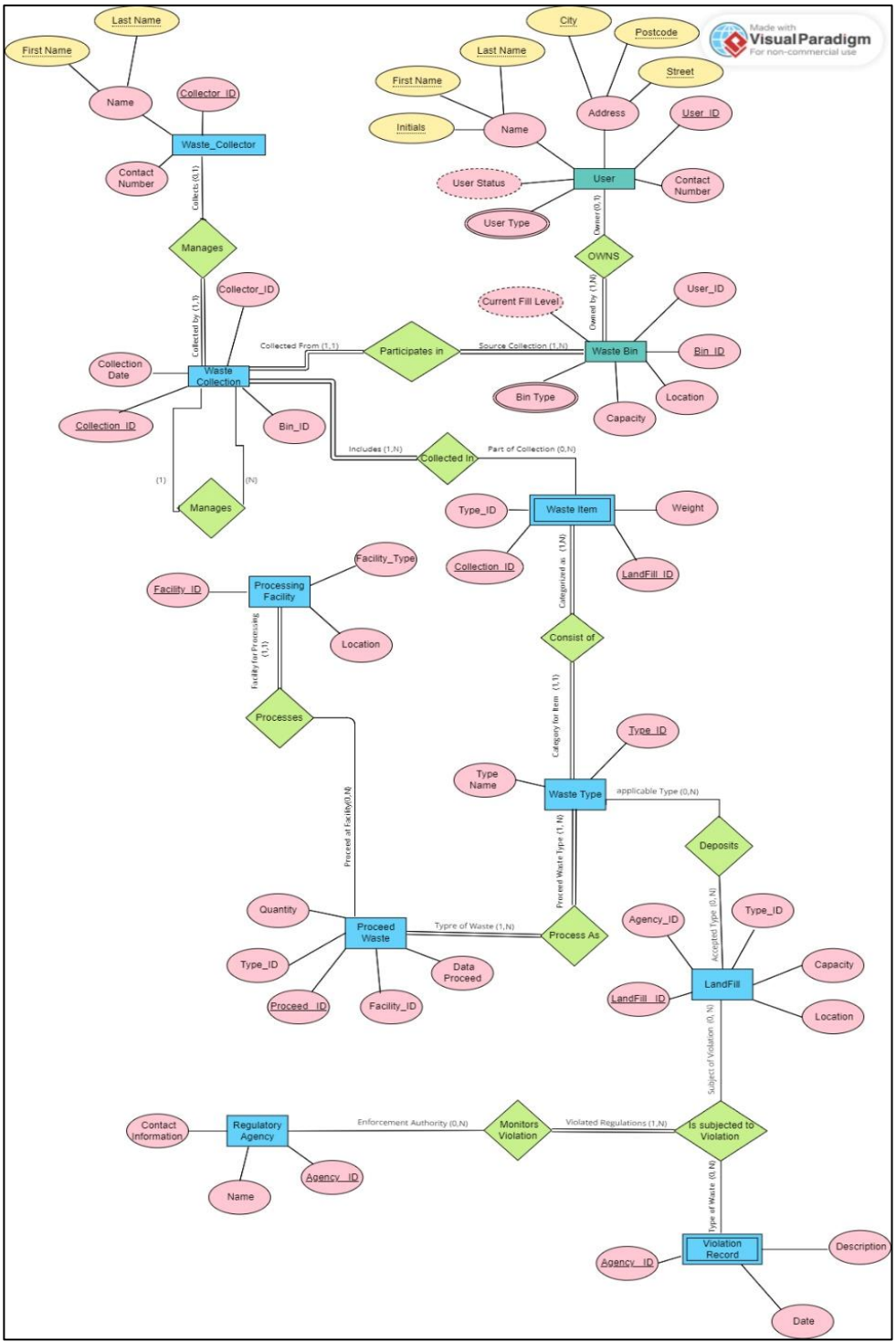


Figure 2-1: ER Diagram of the E-Waste Management System

UML Diagram

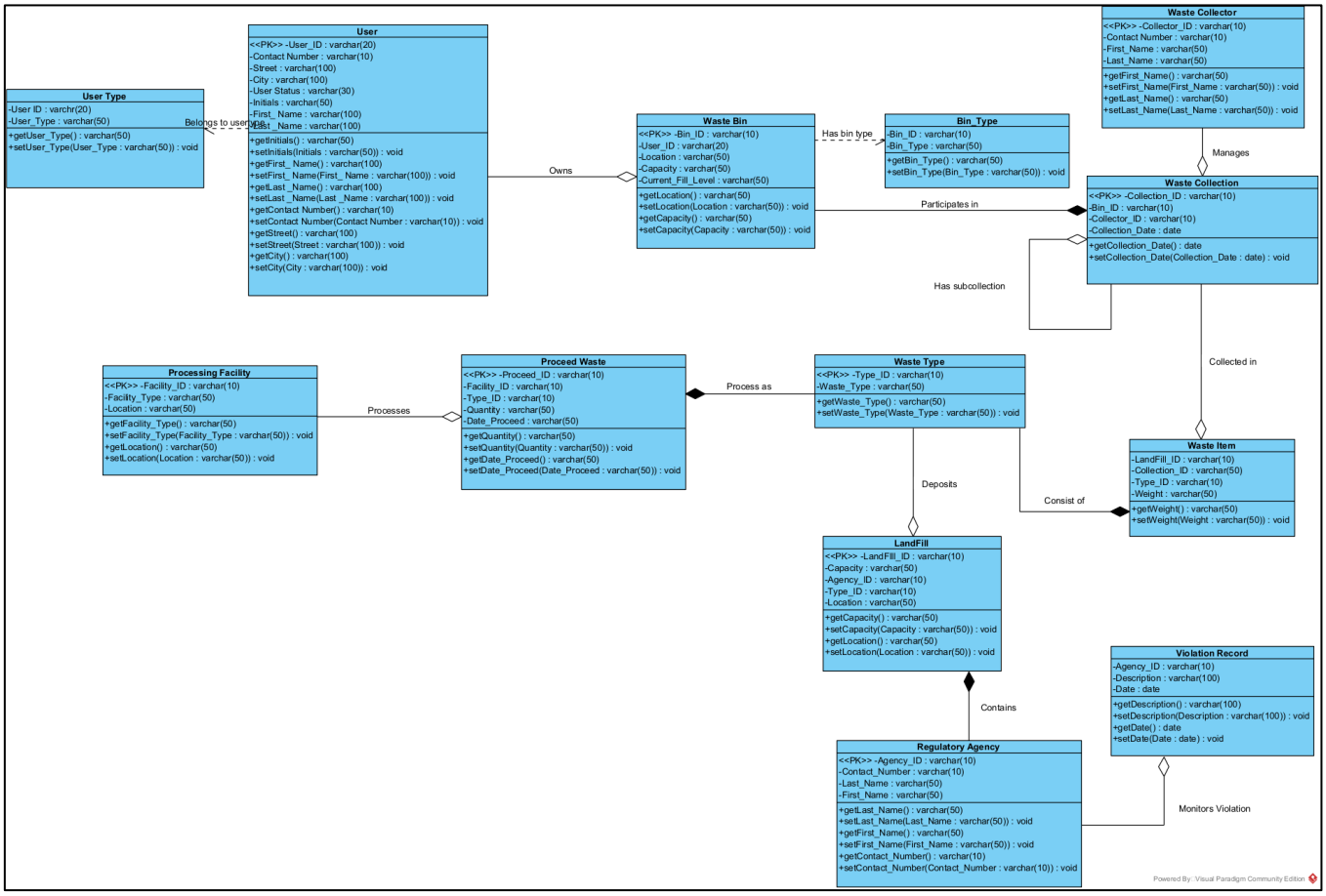
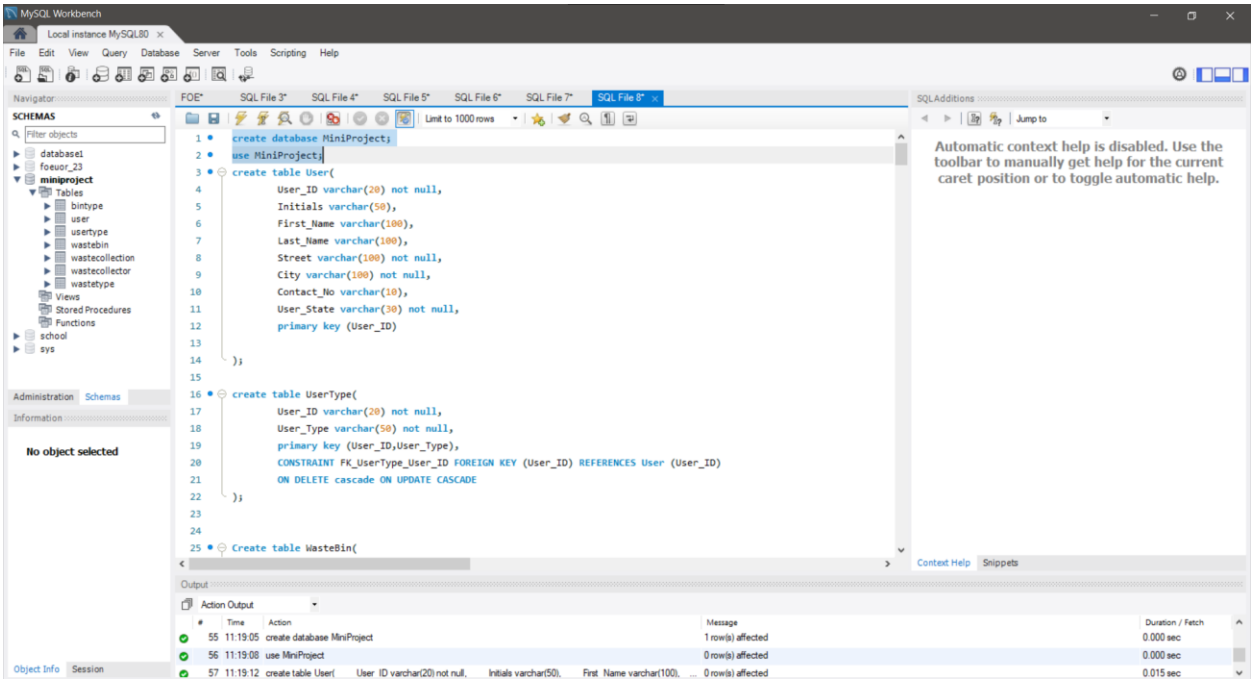


Figure 2-2 : UML Class Diagram of the E-Waste Management System

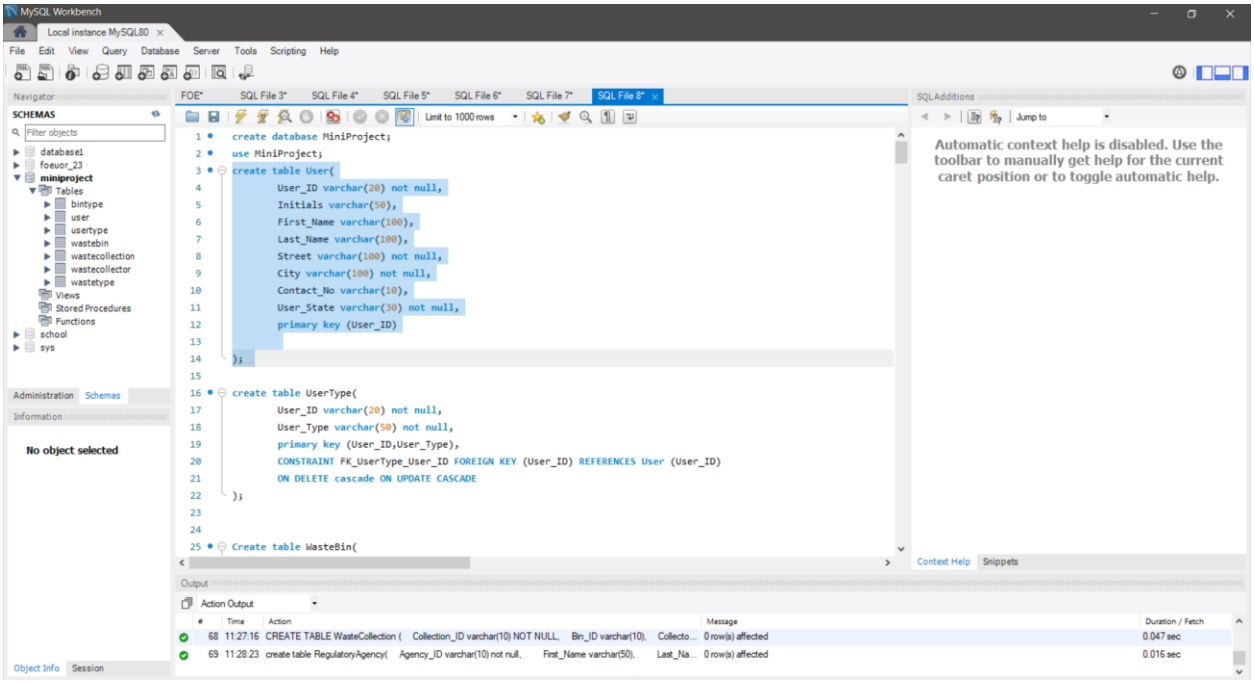
3 Chapter 3- Implementation

Creating and Use Data Base

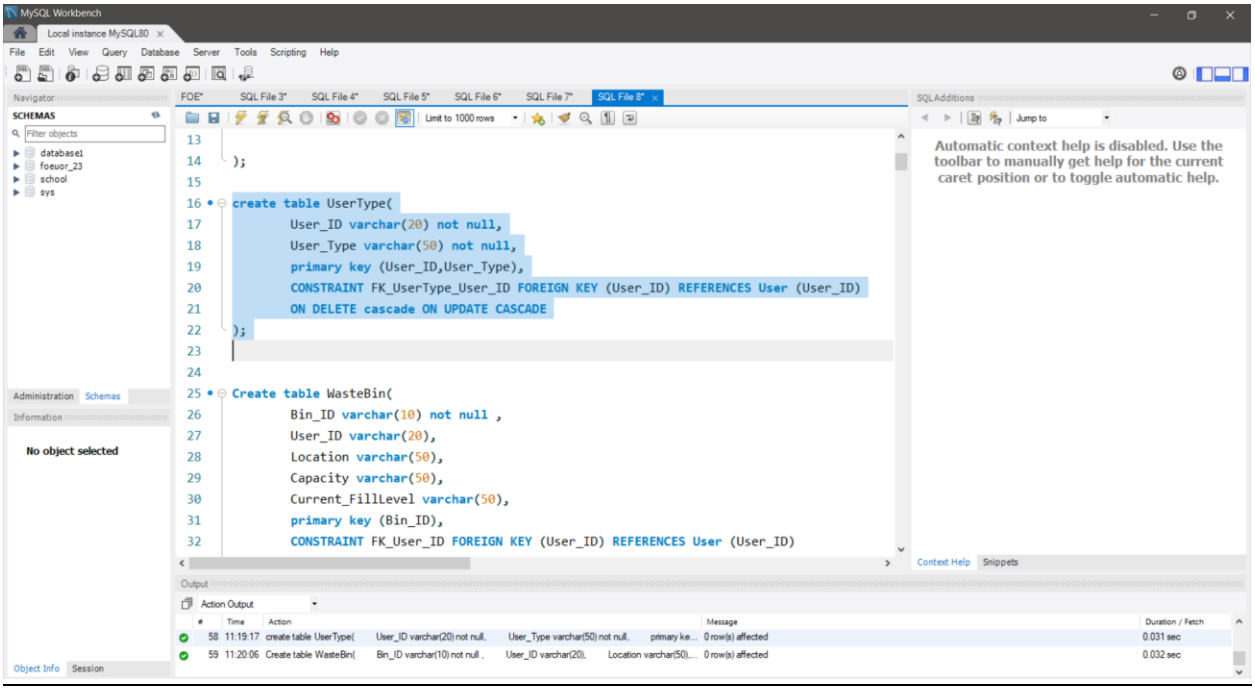


Creating Tables

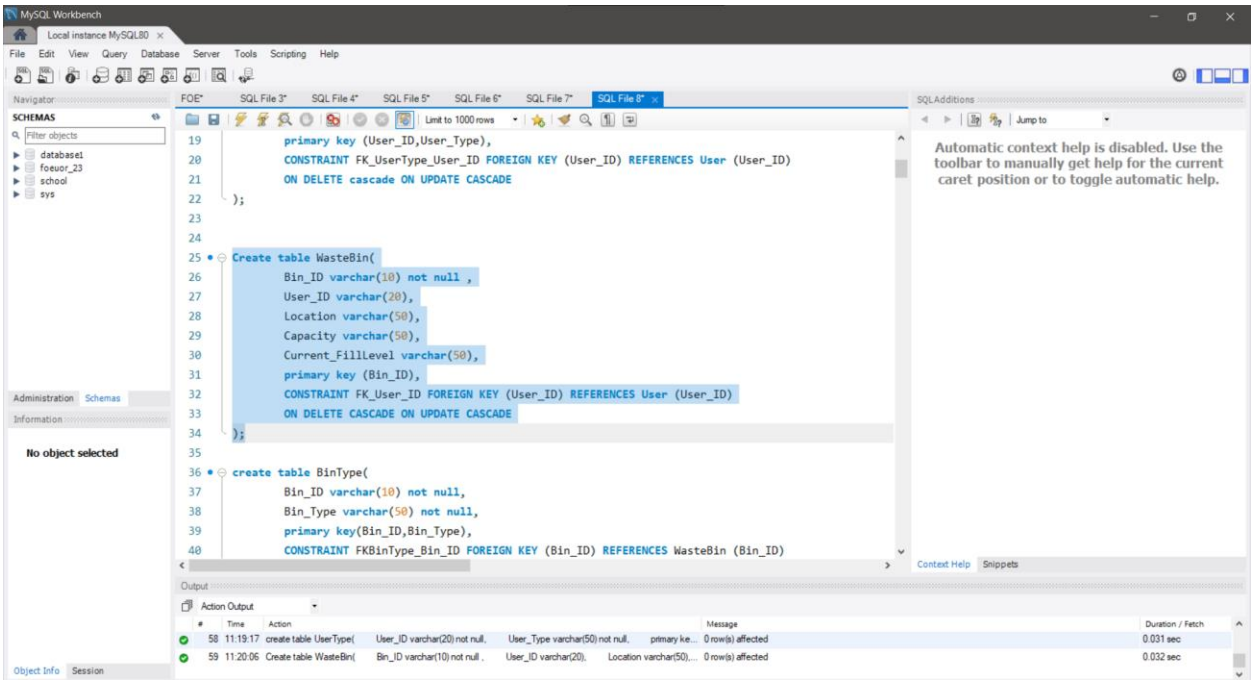
User Table



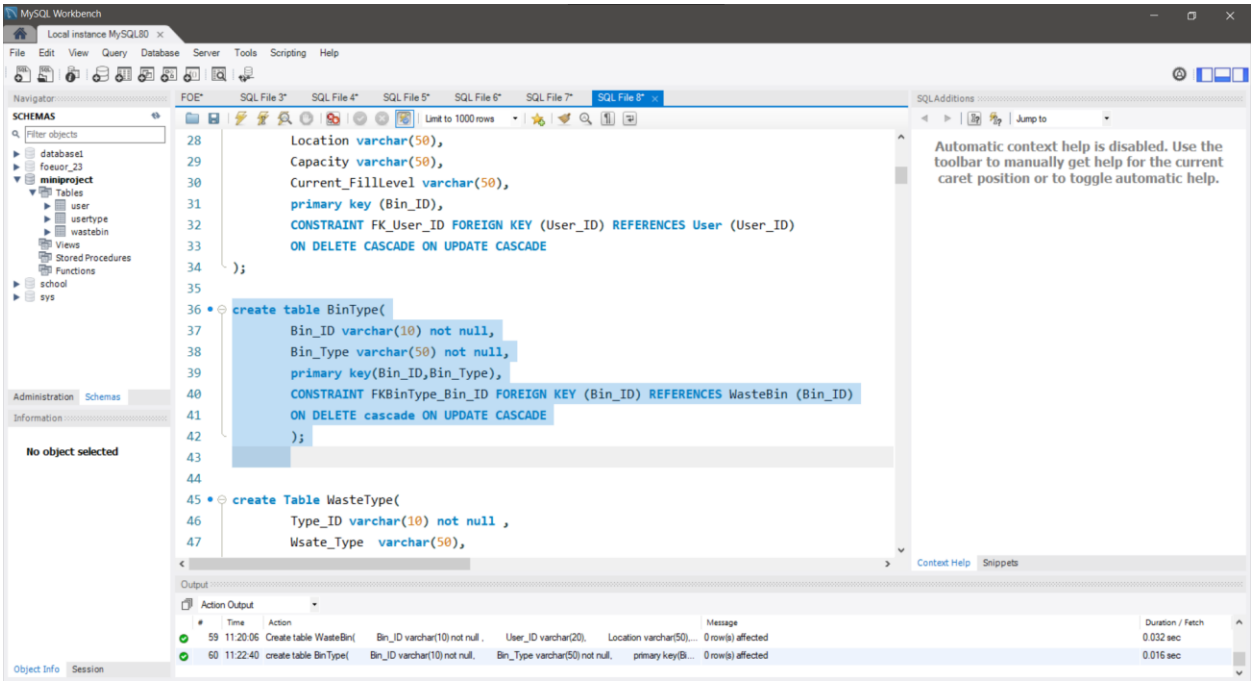
UserType Table



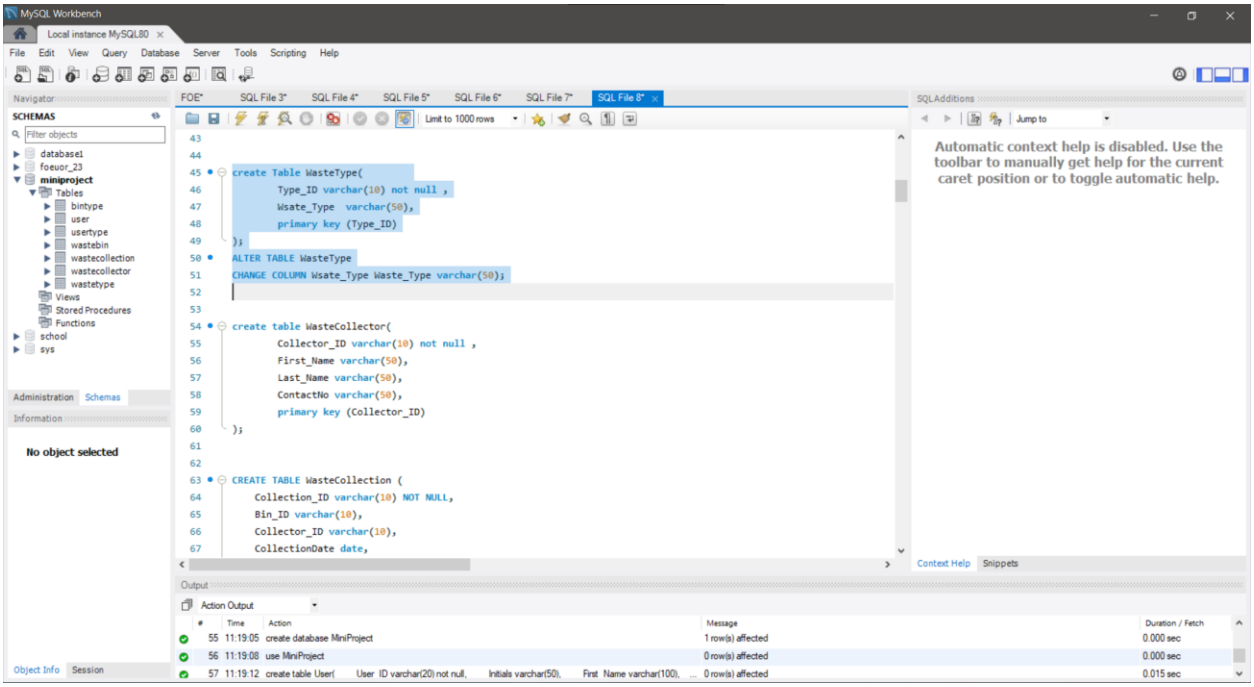
WasteBin Table



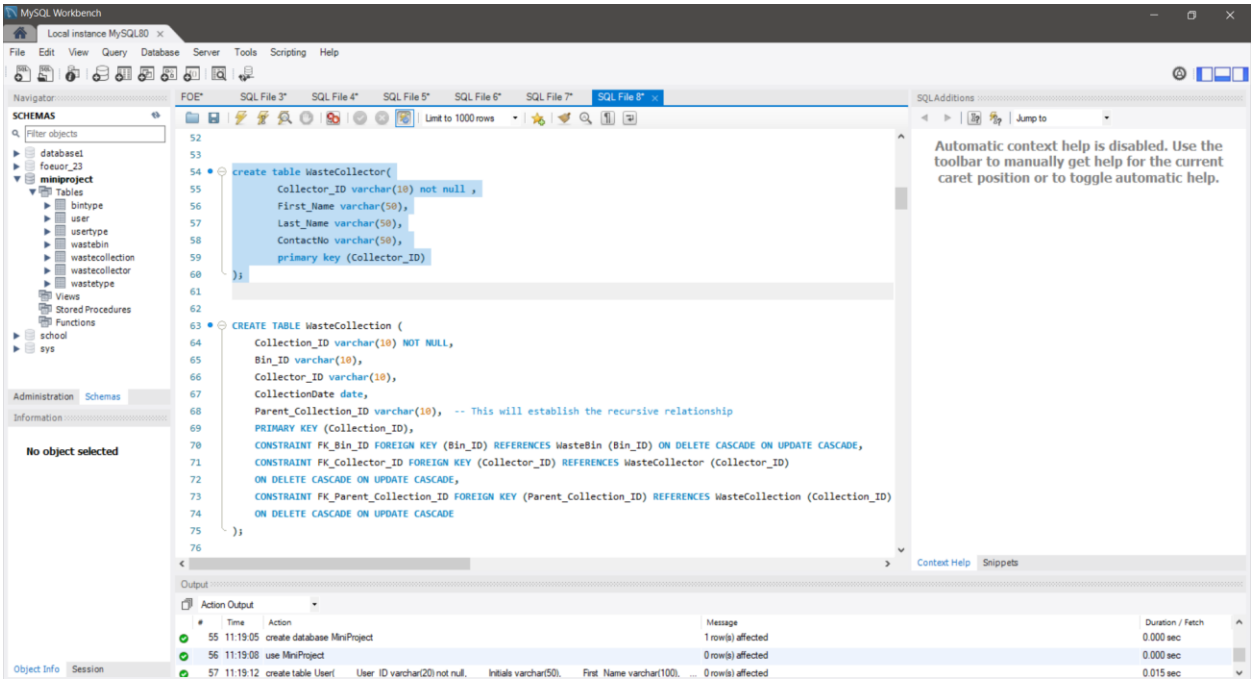
BinType Table



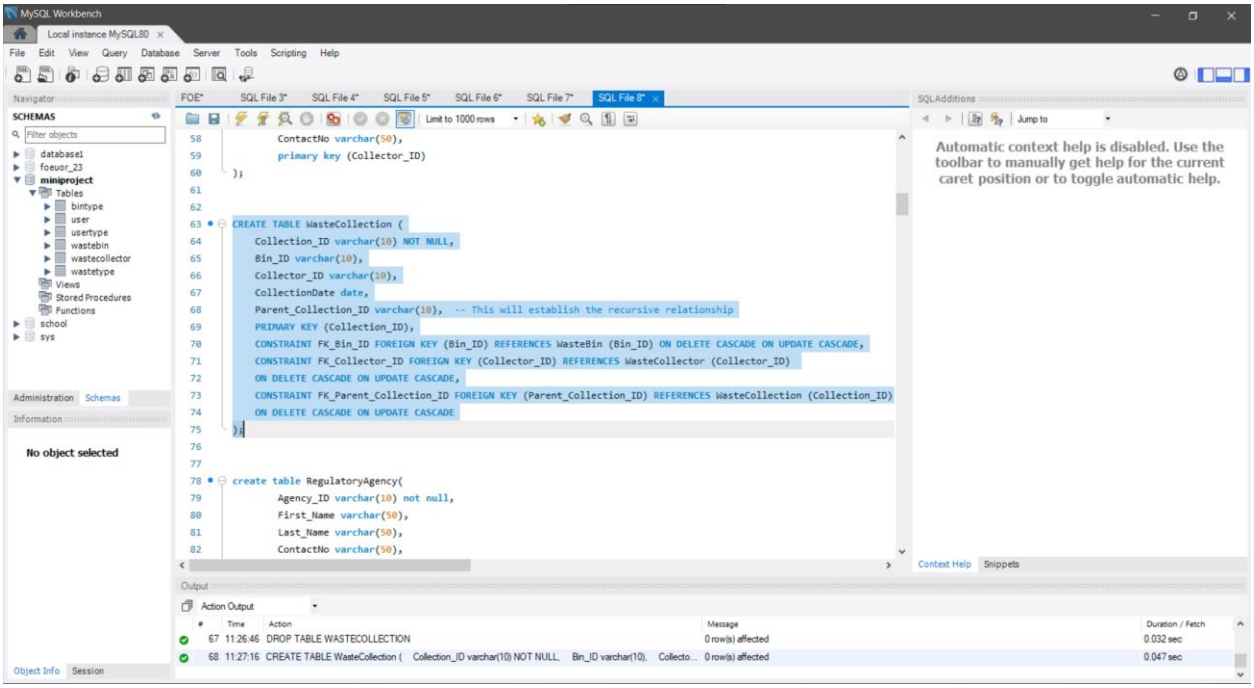
WasteType Table



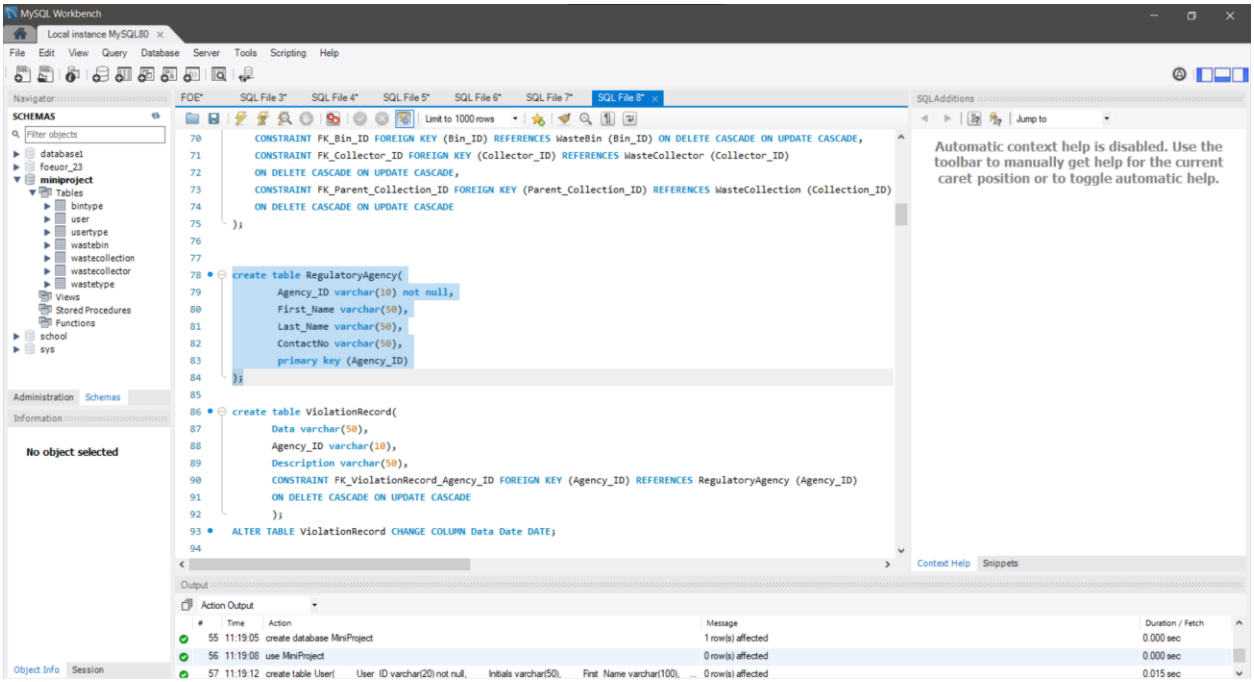
WasteCollector Table



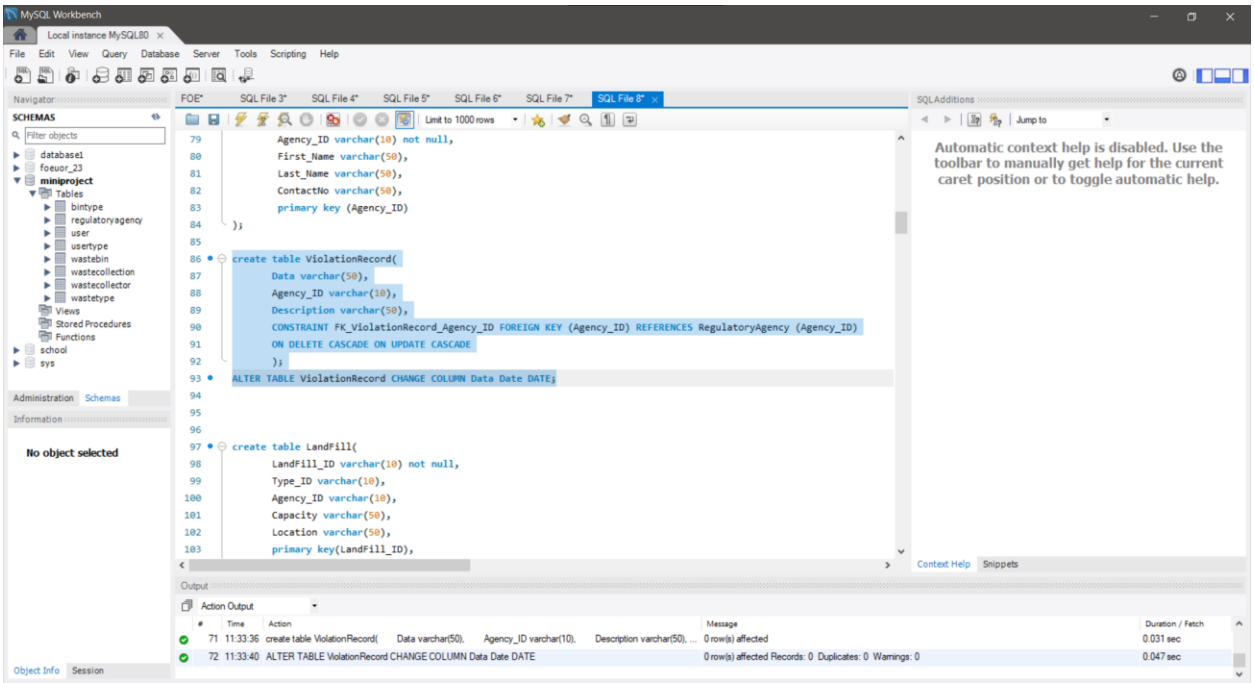
WasteCollection Table



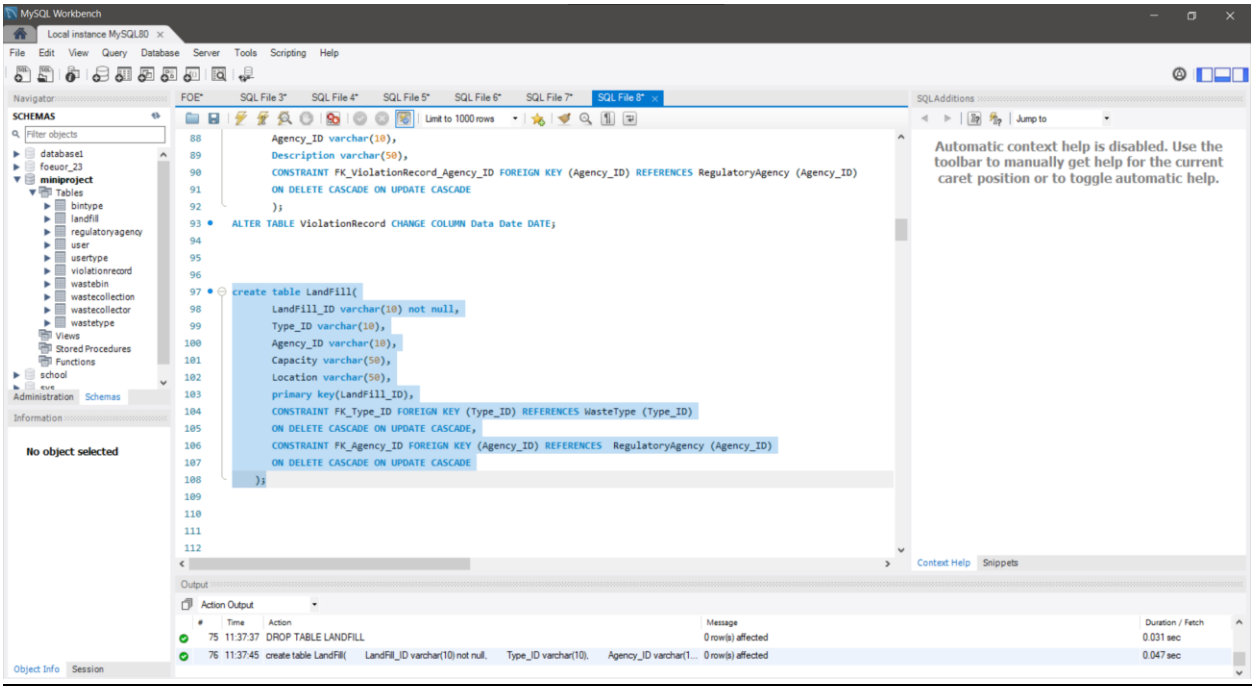
RegulatoryAgency Table



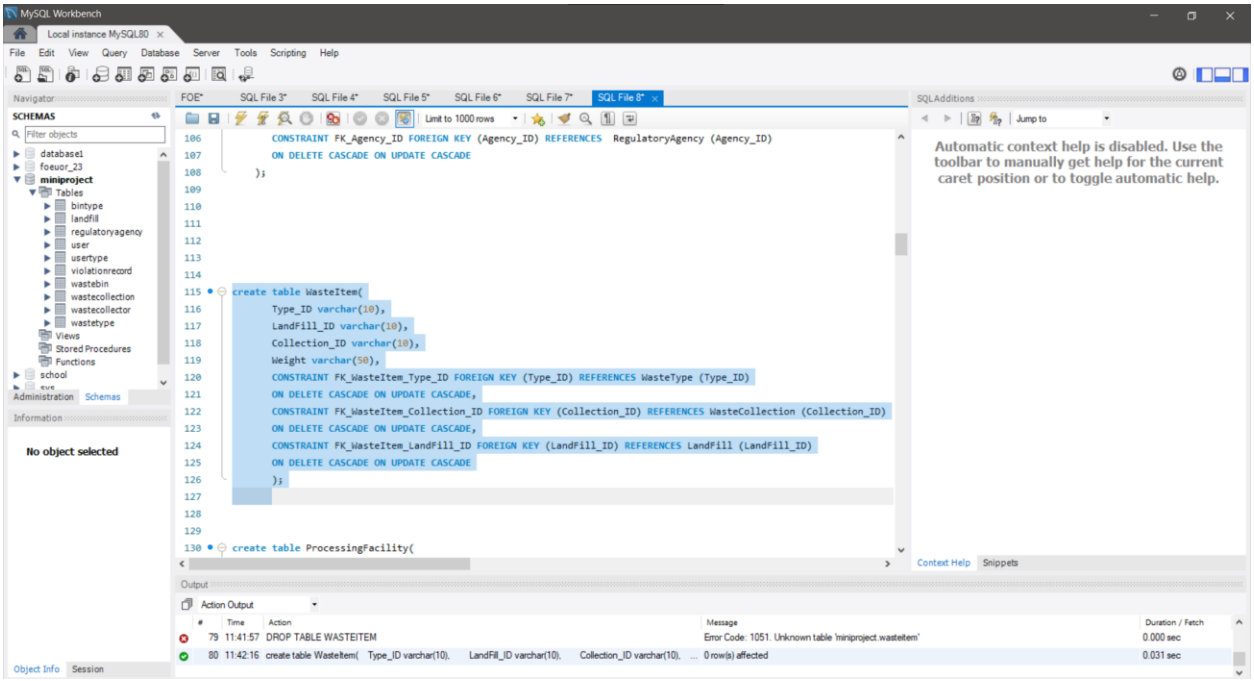
ViolationRecord Table



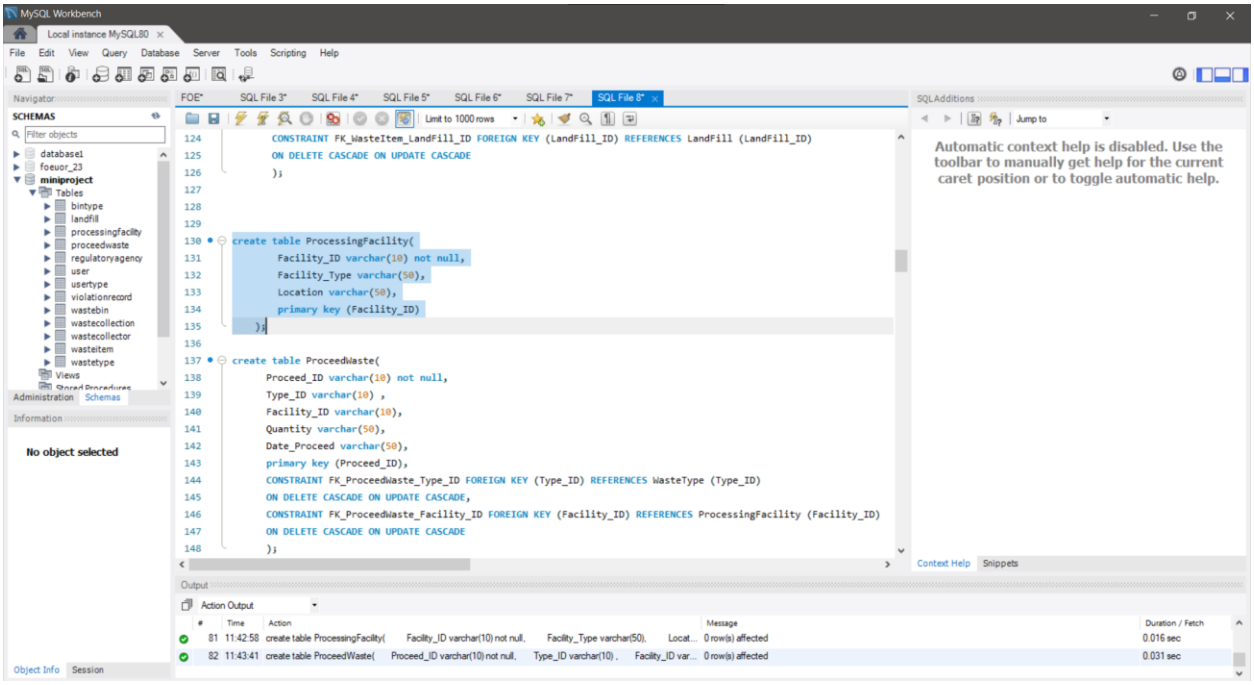
LandFill Table



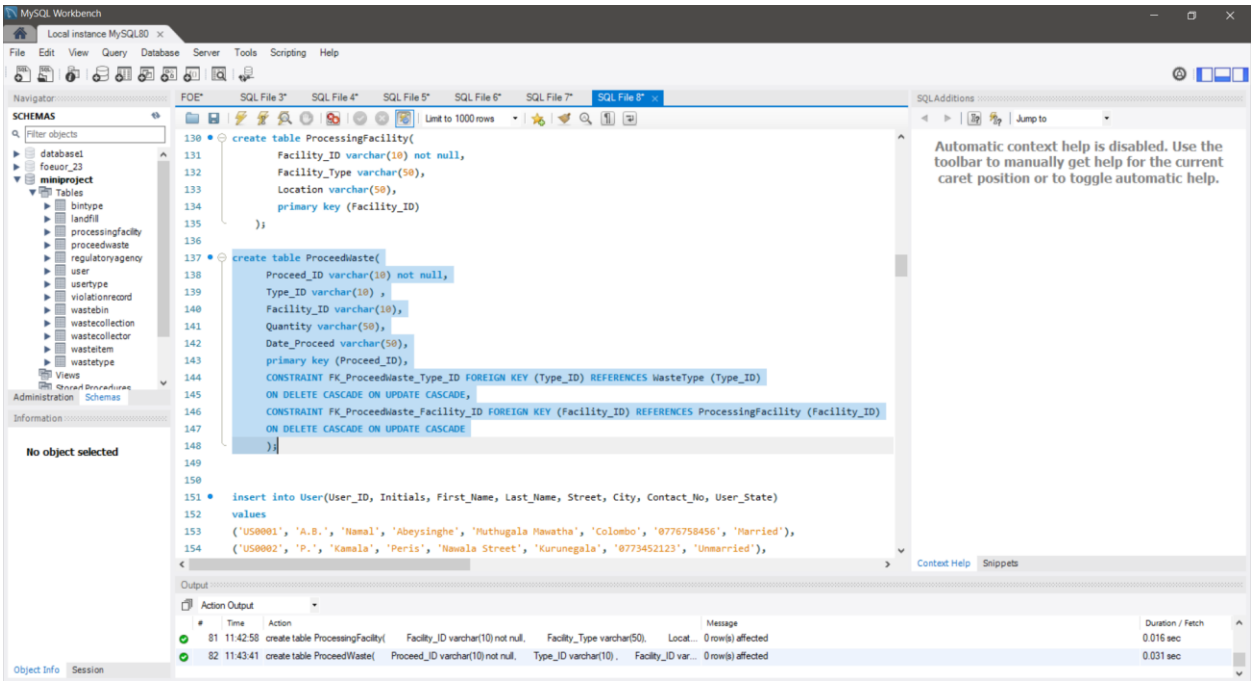
WasteItem Table



ProcessingFacility Table

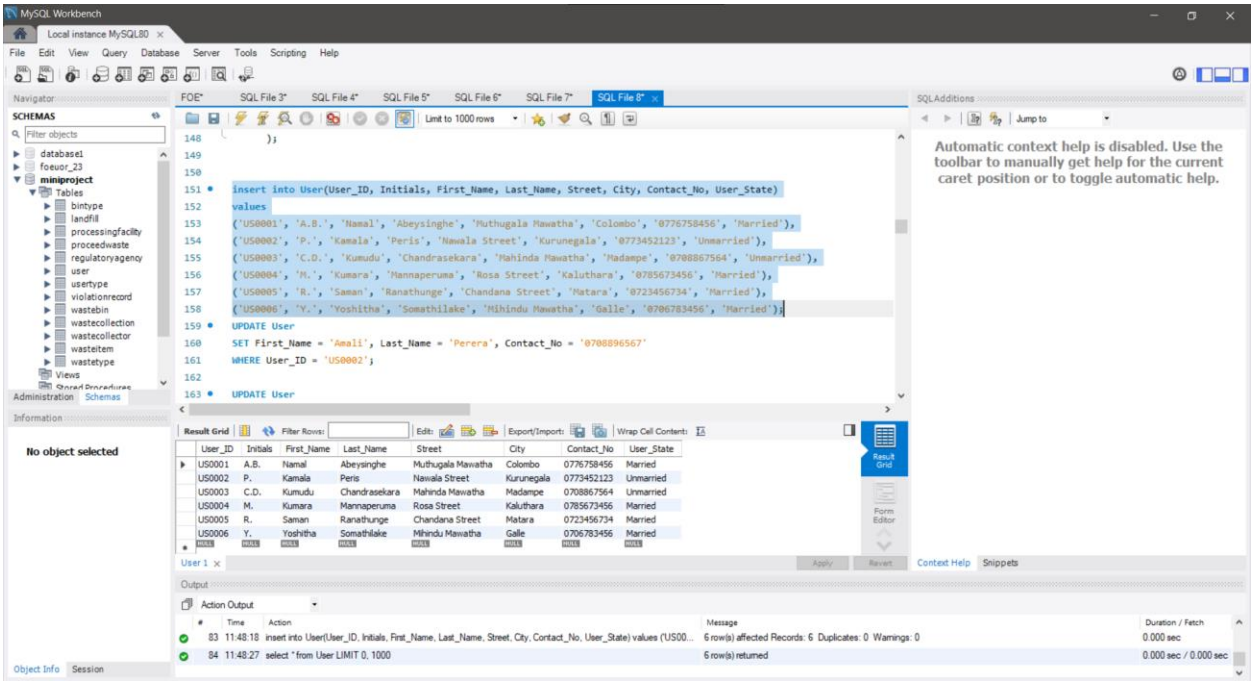


ProceedWaste

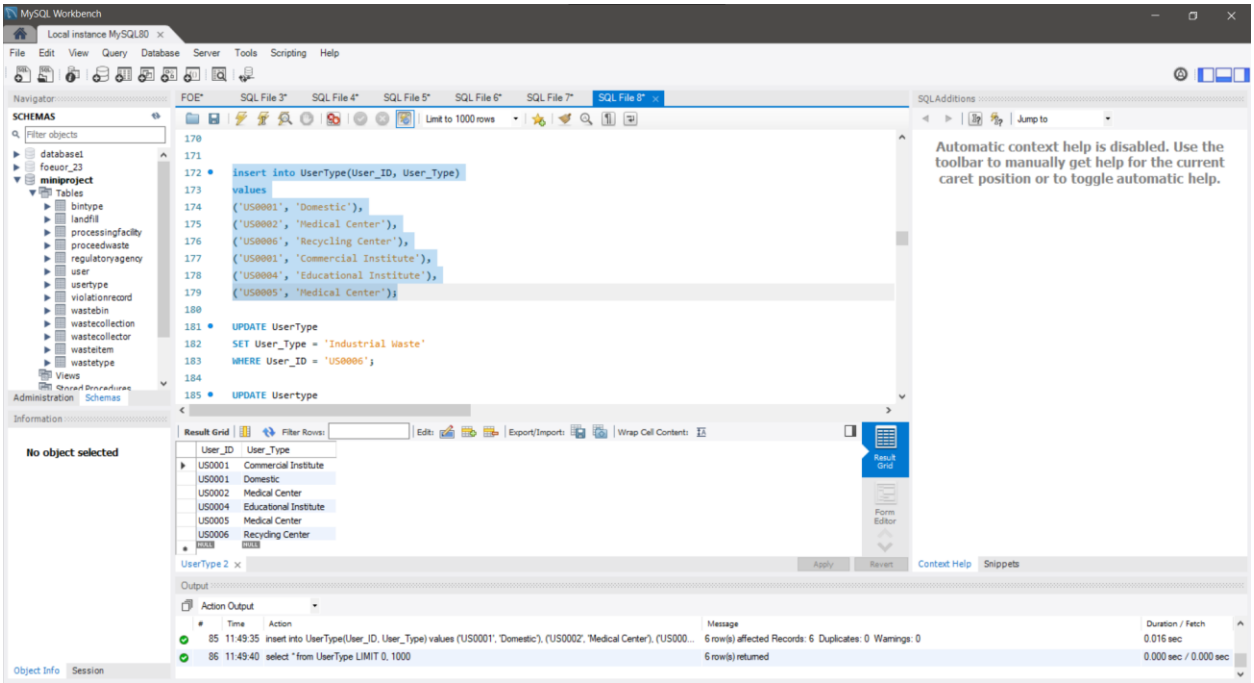


Inserting Data in to tables

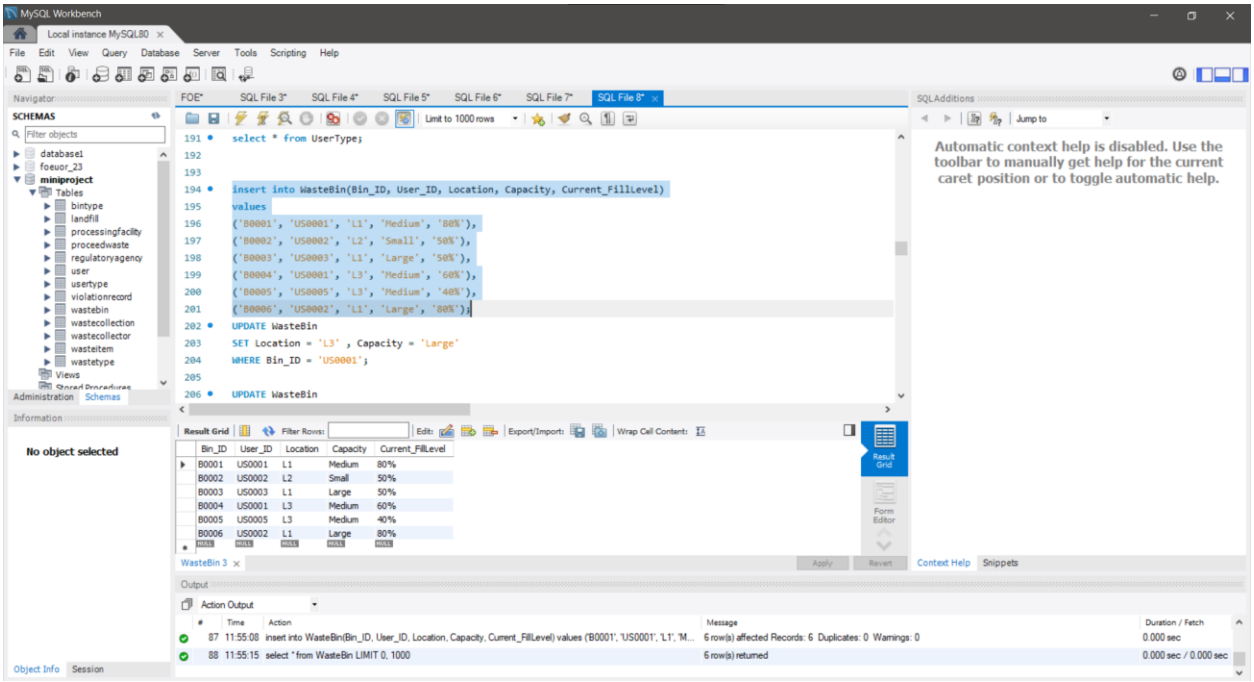
User Table



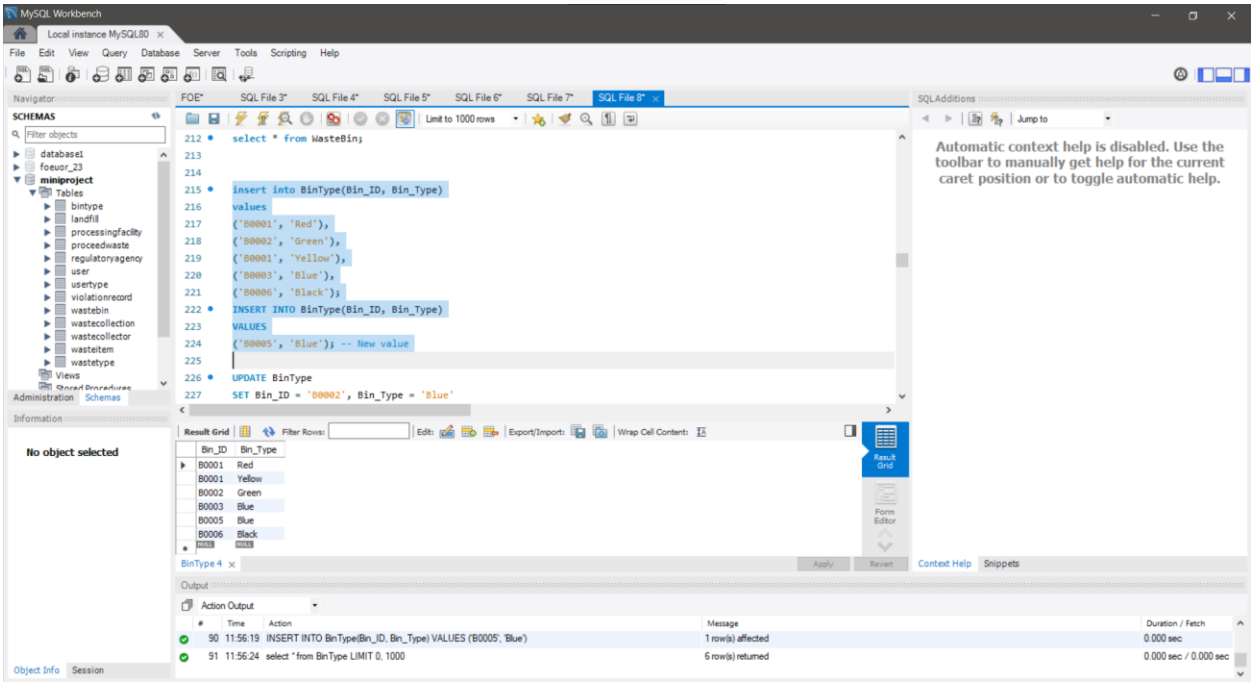
UserType Table



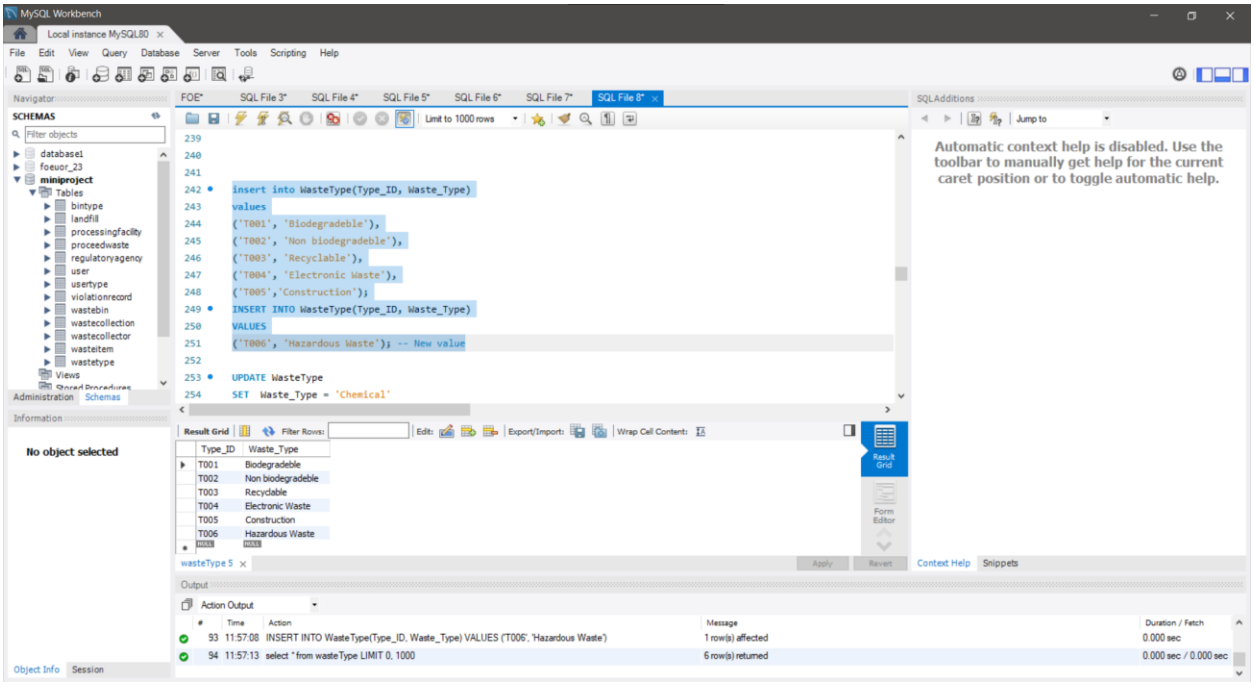
WasteBin Table



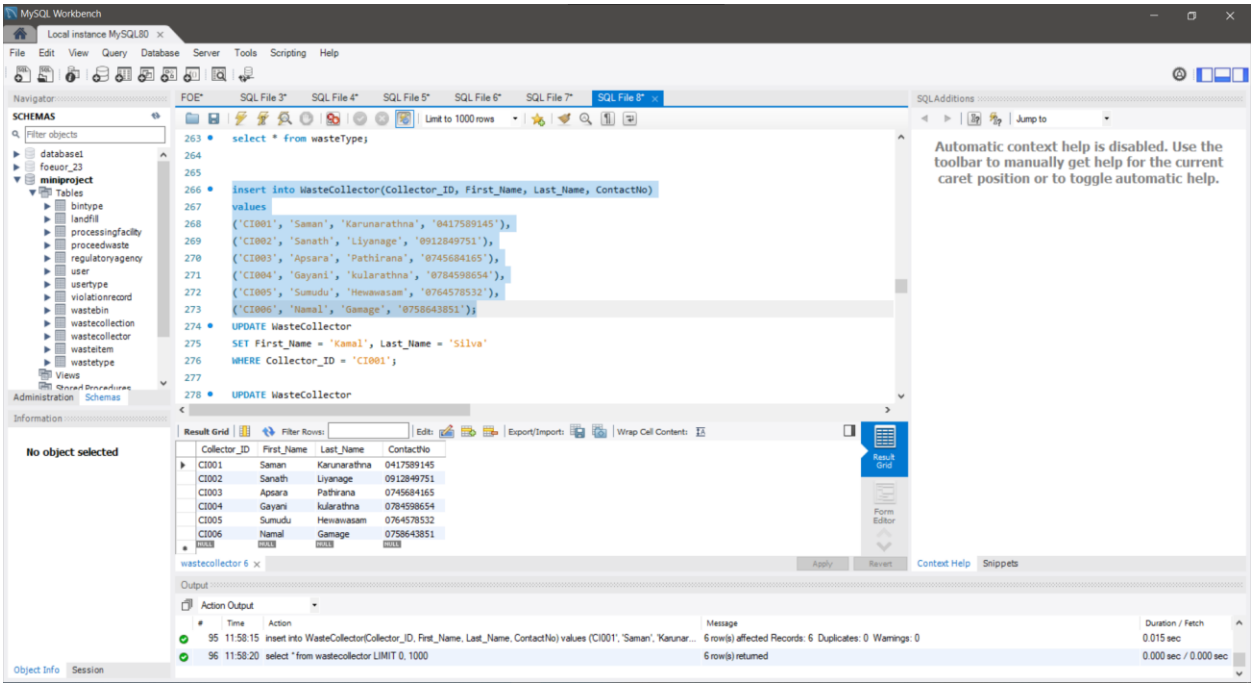
BinType Table



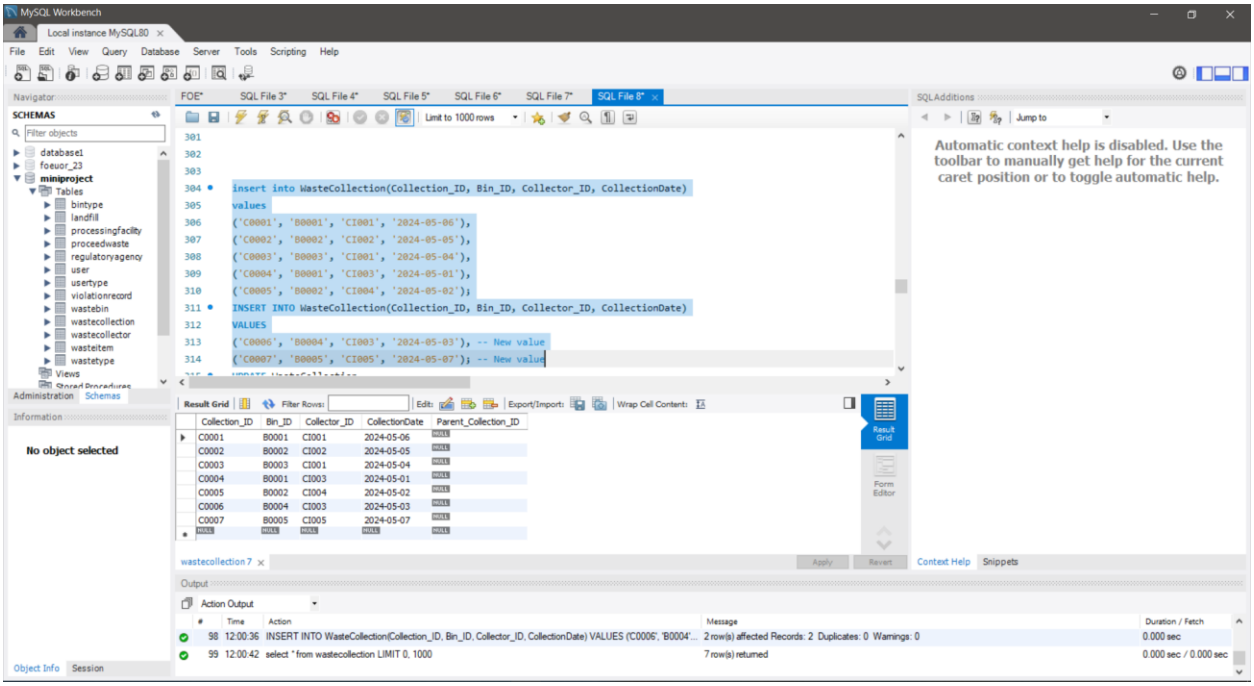
WasteType Table



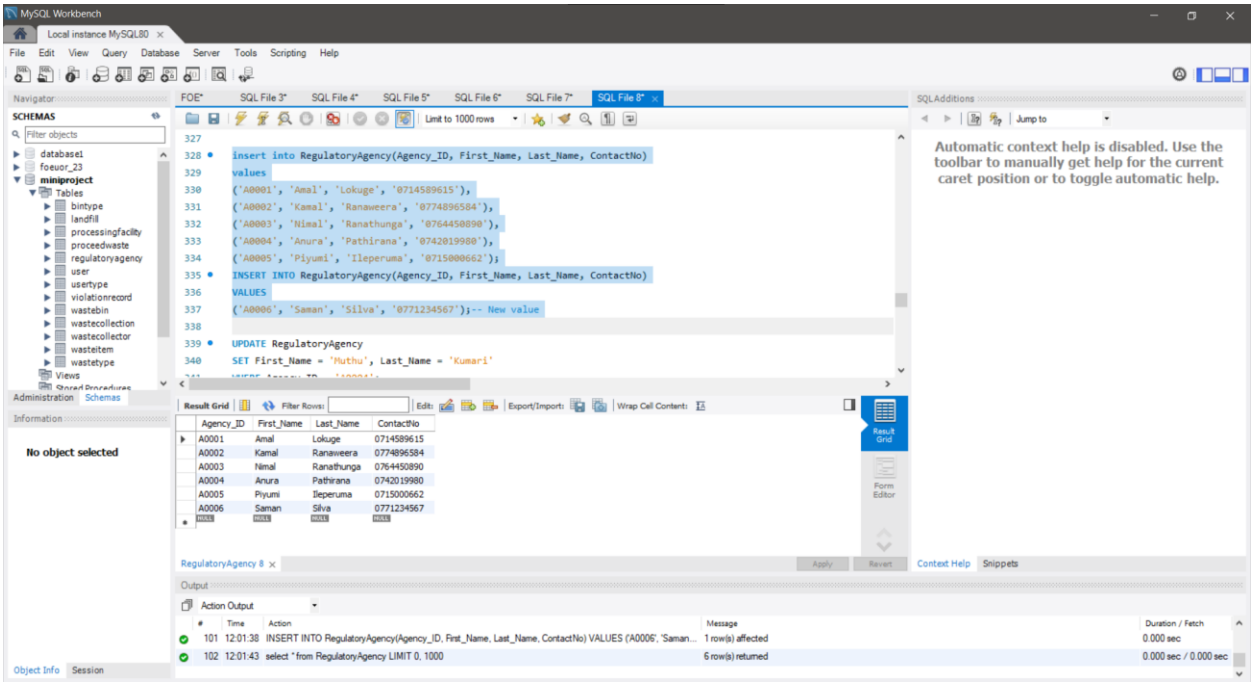
WasteCollector Table



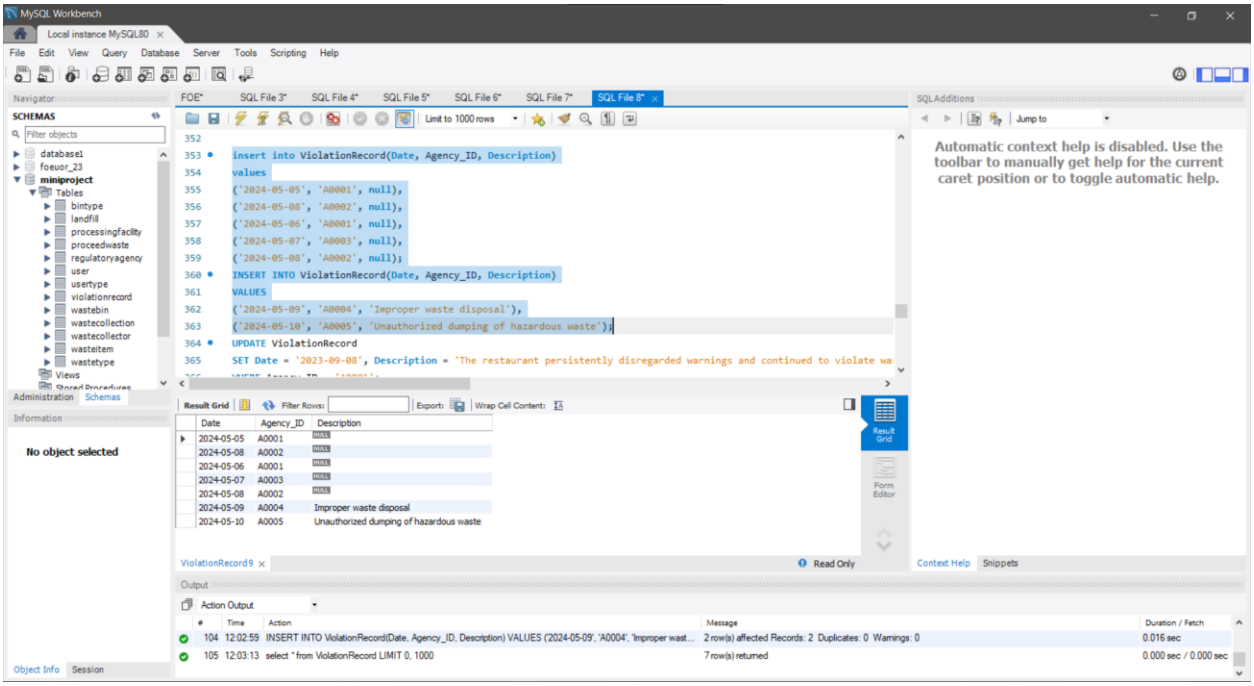
WasteCollection Table



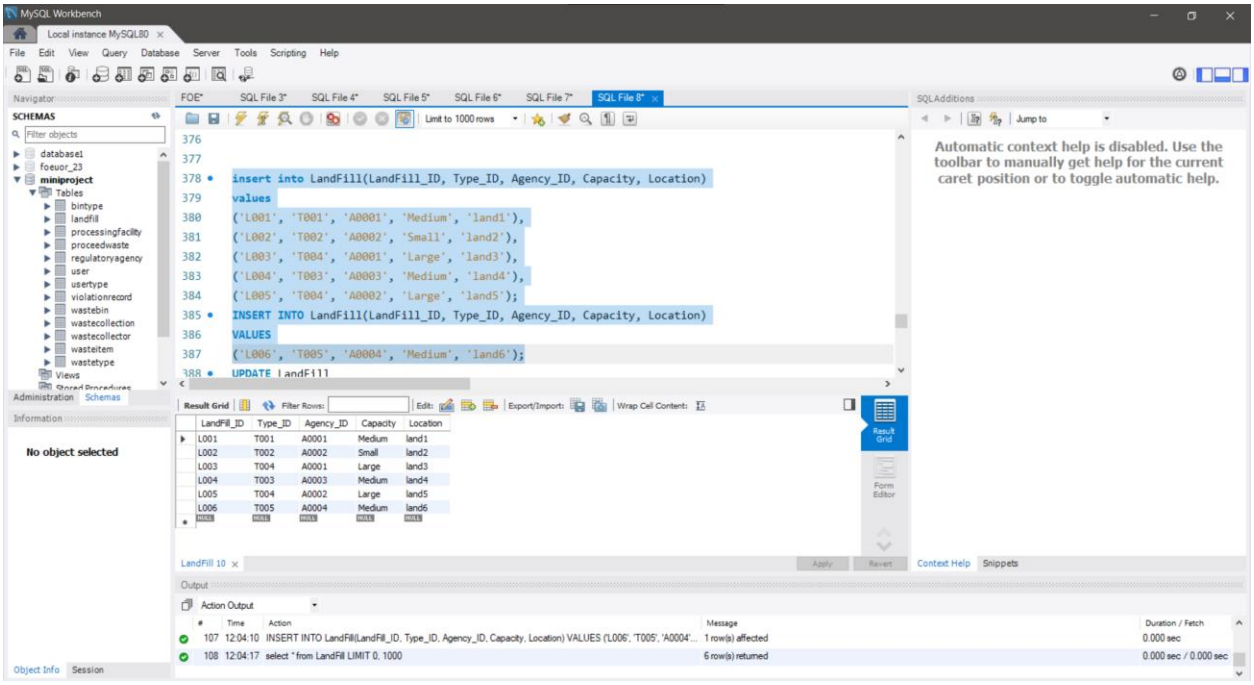
RegulatoryAgency Table



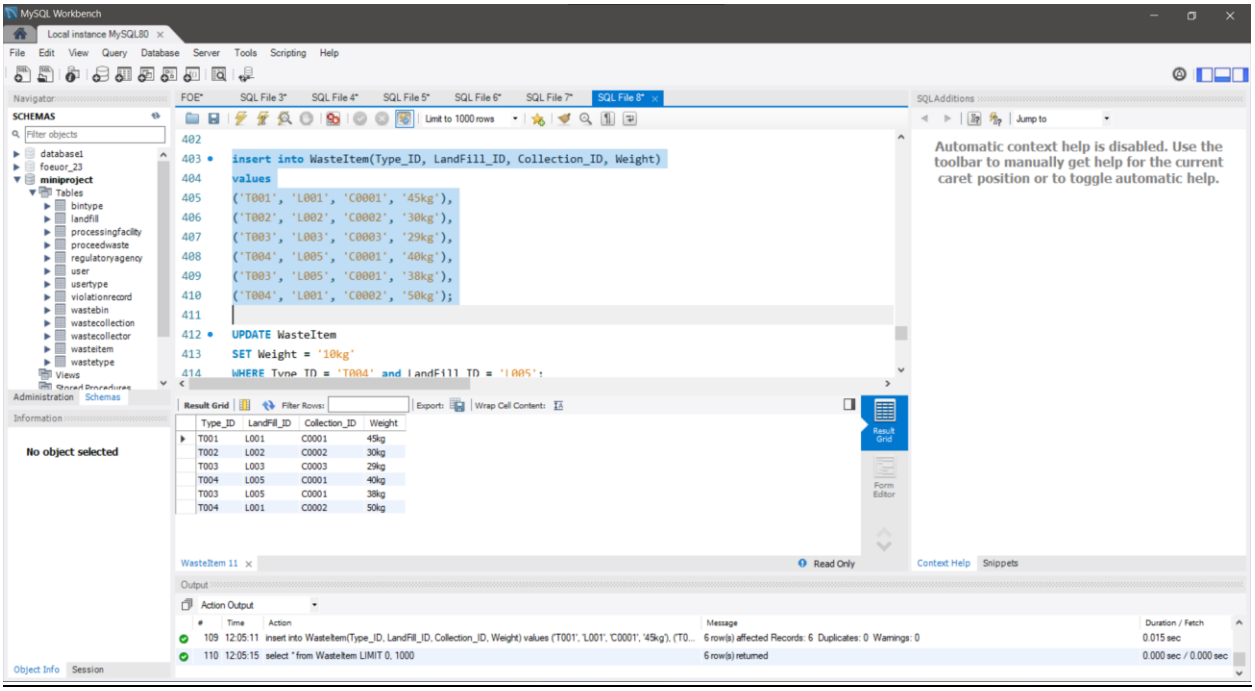
ViolationRecord Table



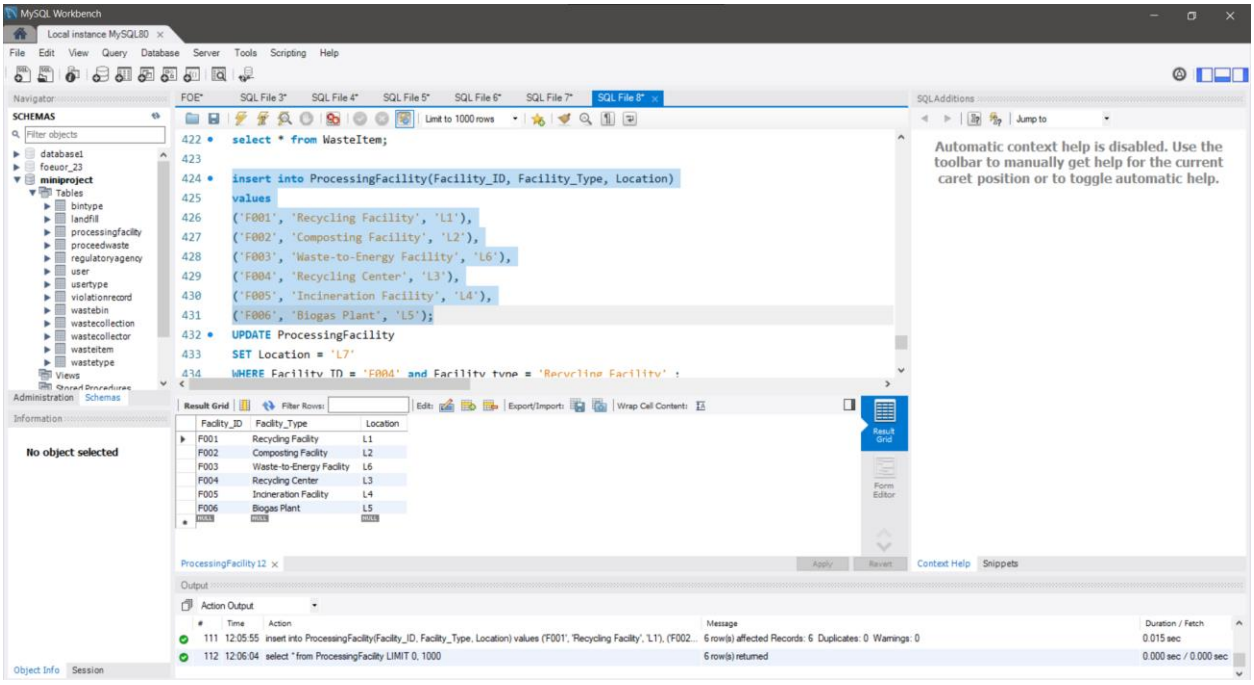
LandFill Table



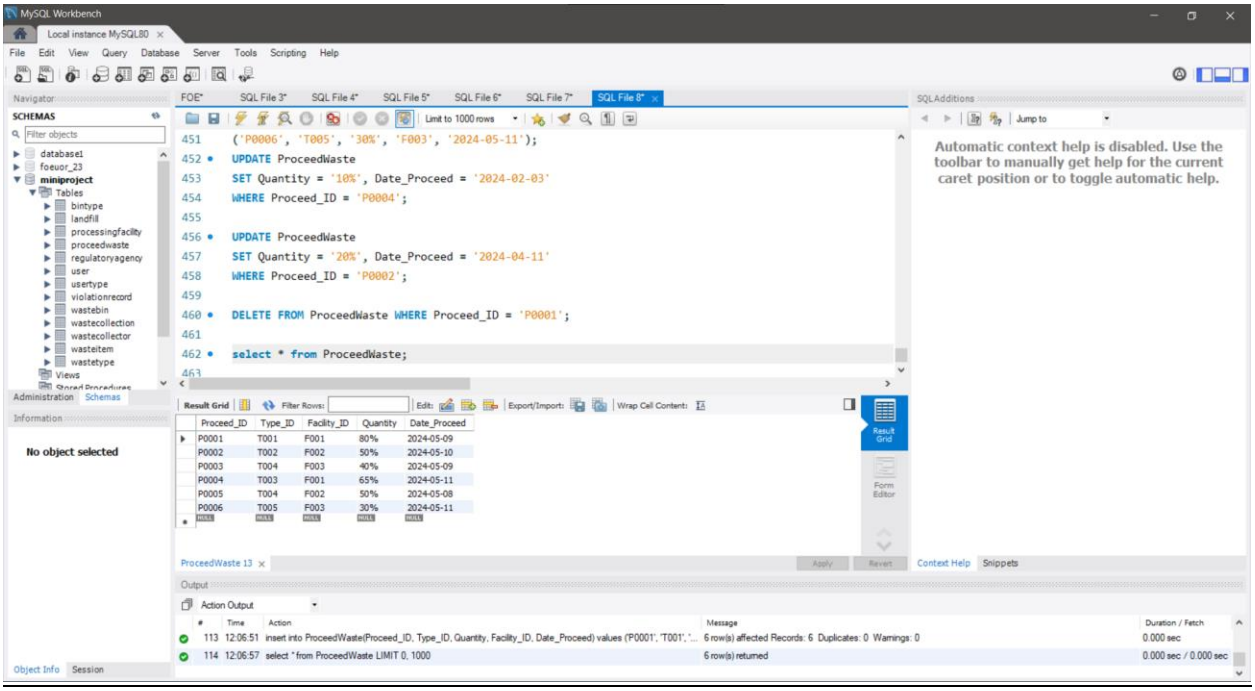
WasteItem Table



ProcessingFacility Table

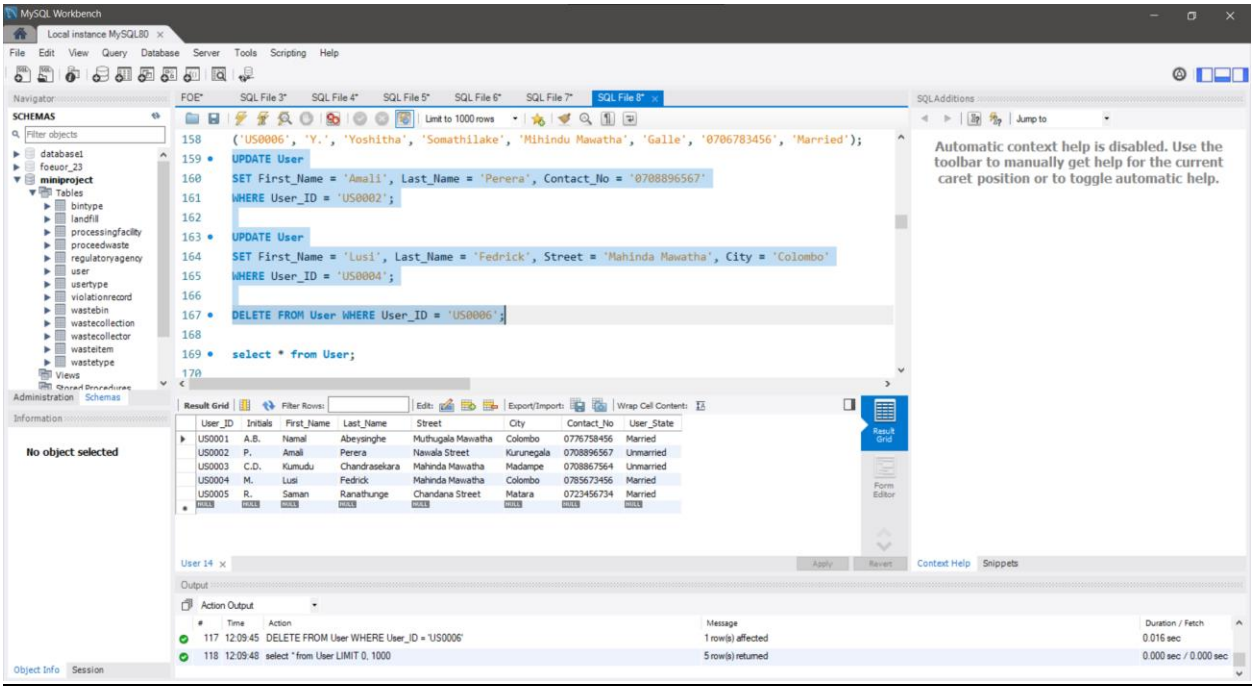


ProceedWaste Table

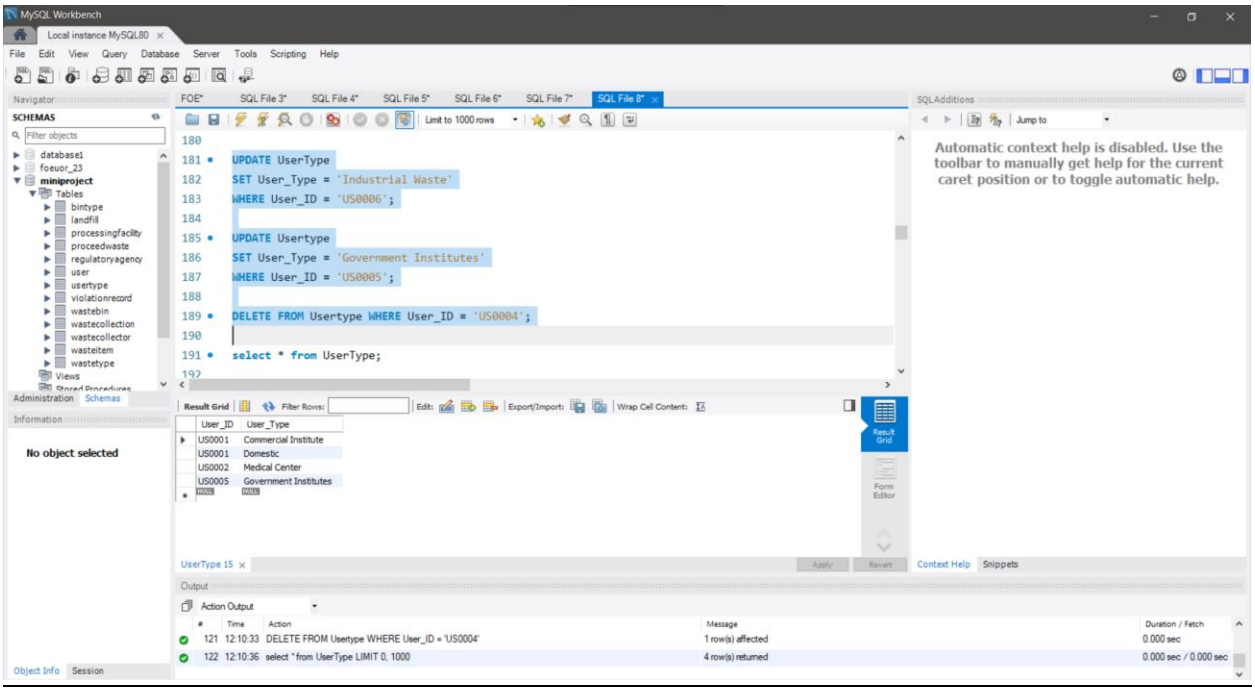


Delete and Update Tables

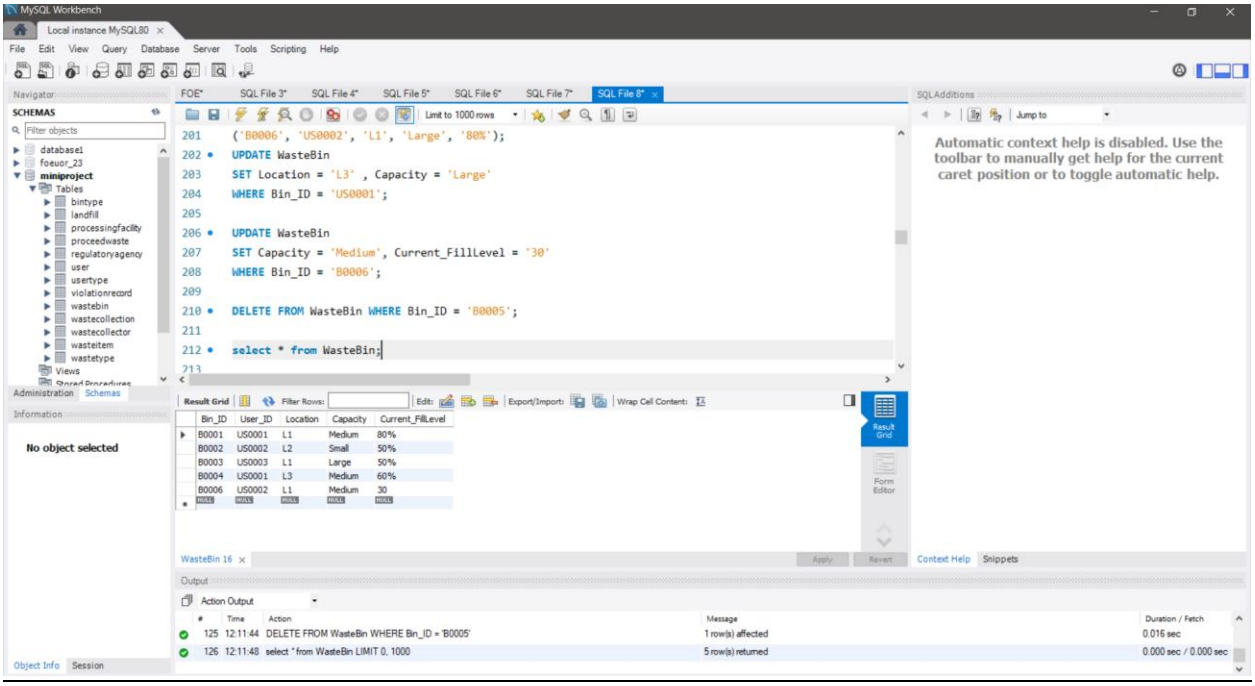
User Table



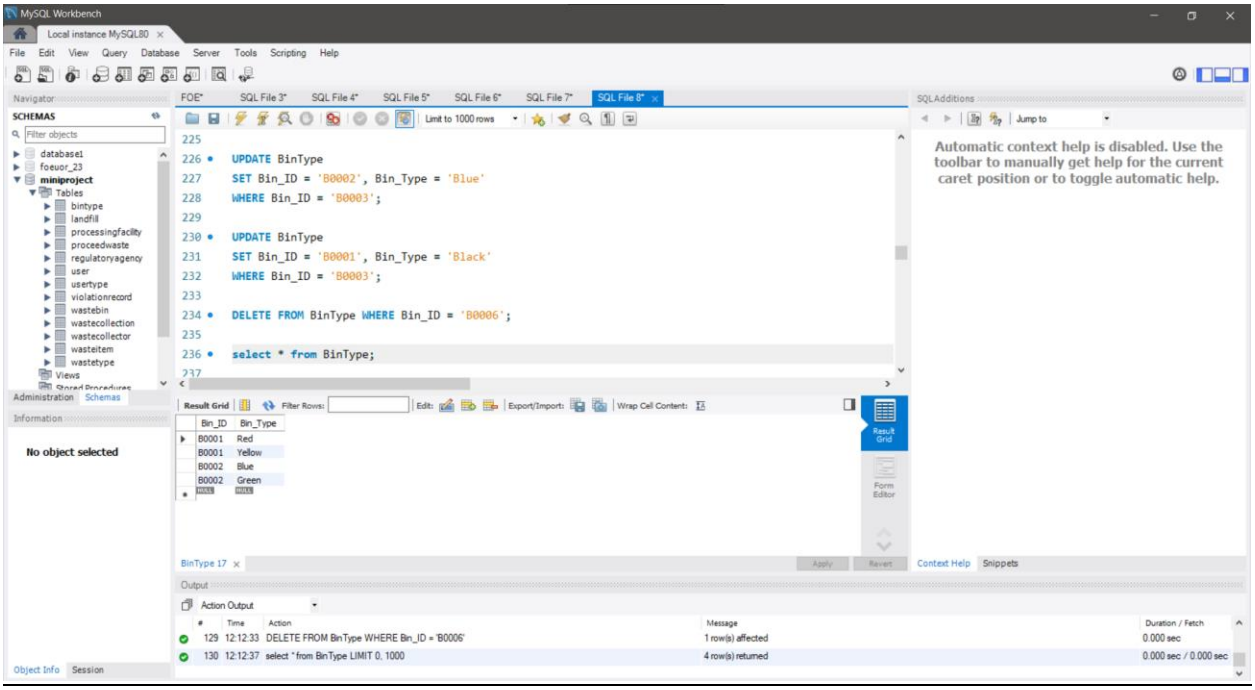
UserType Table



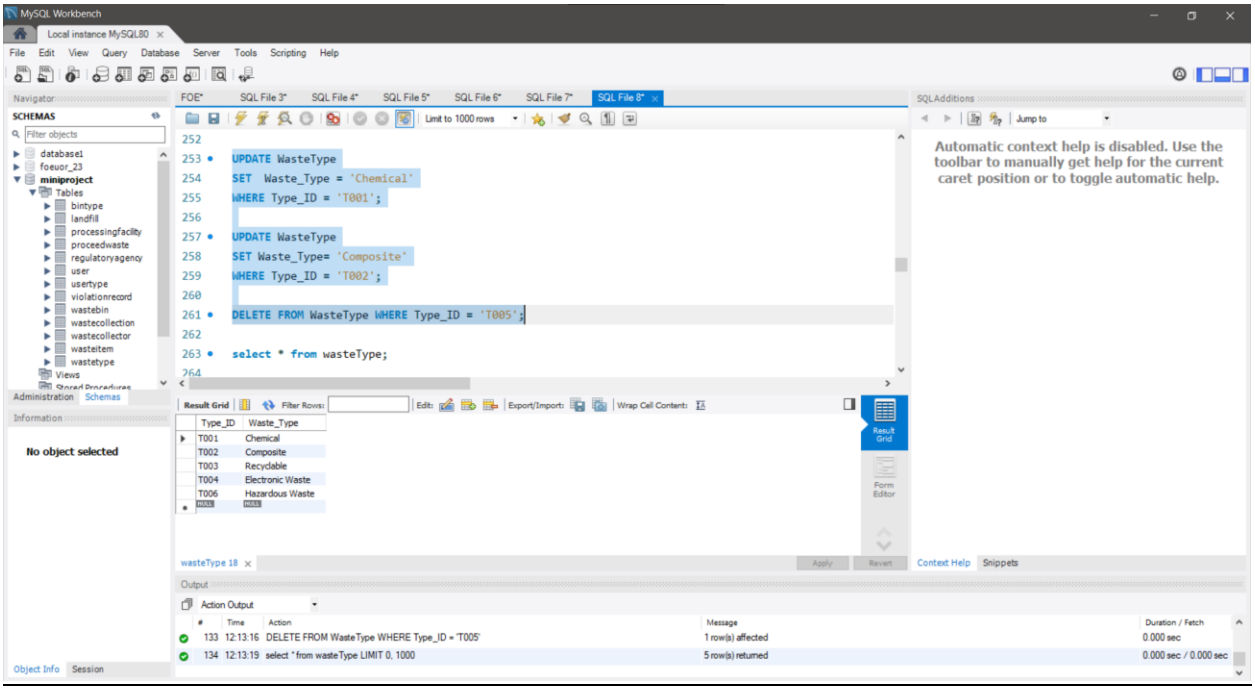
WasteBin Table



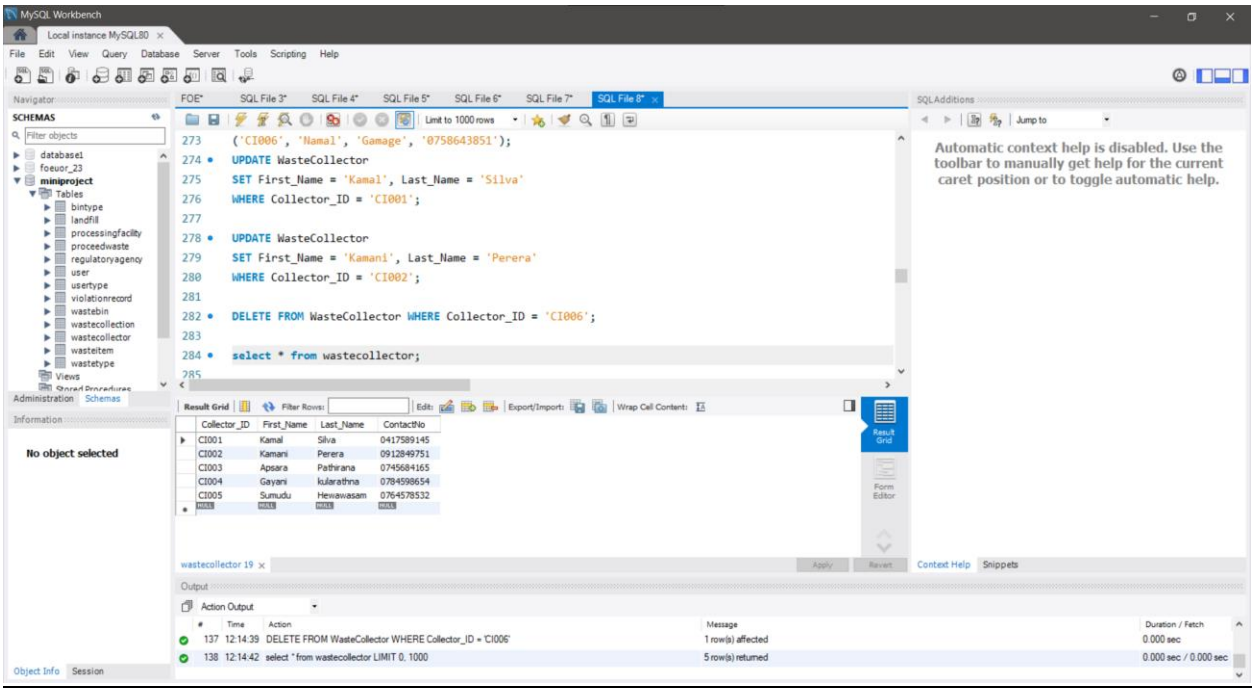
BinType Table



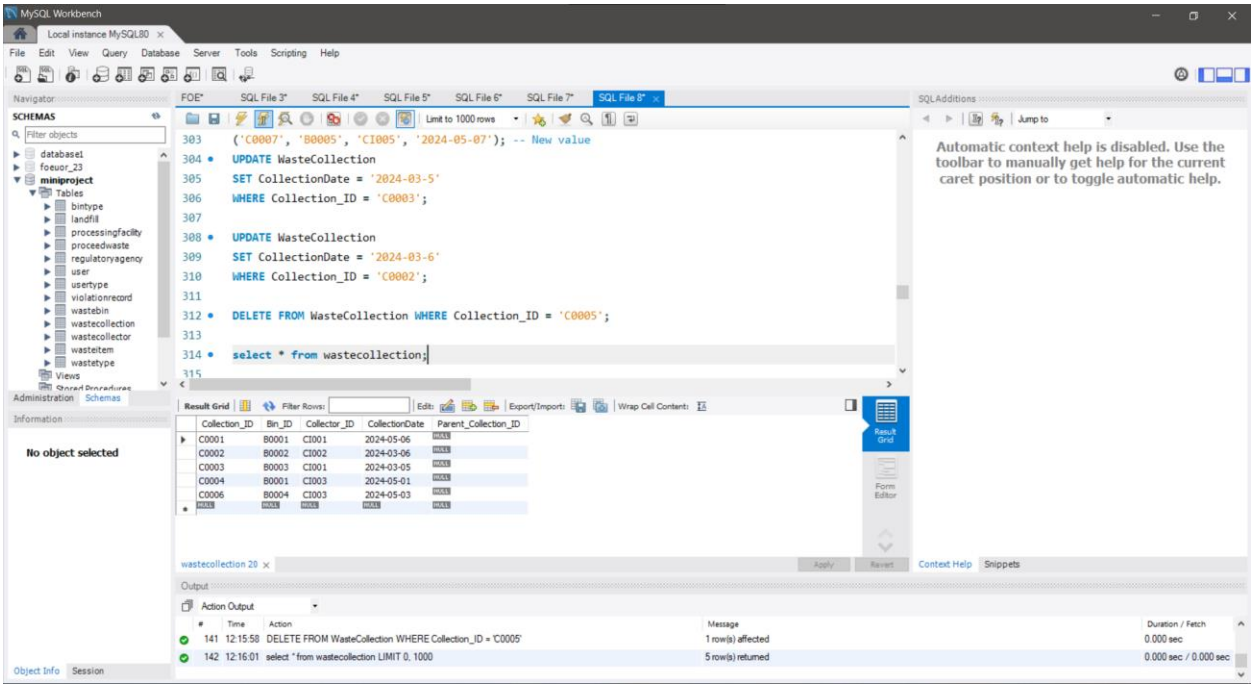
WasteType Table



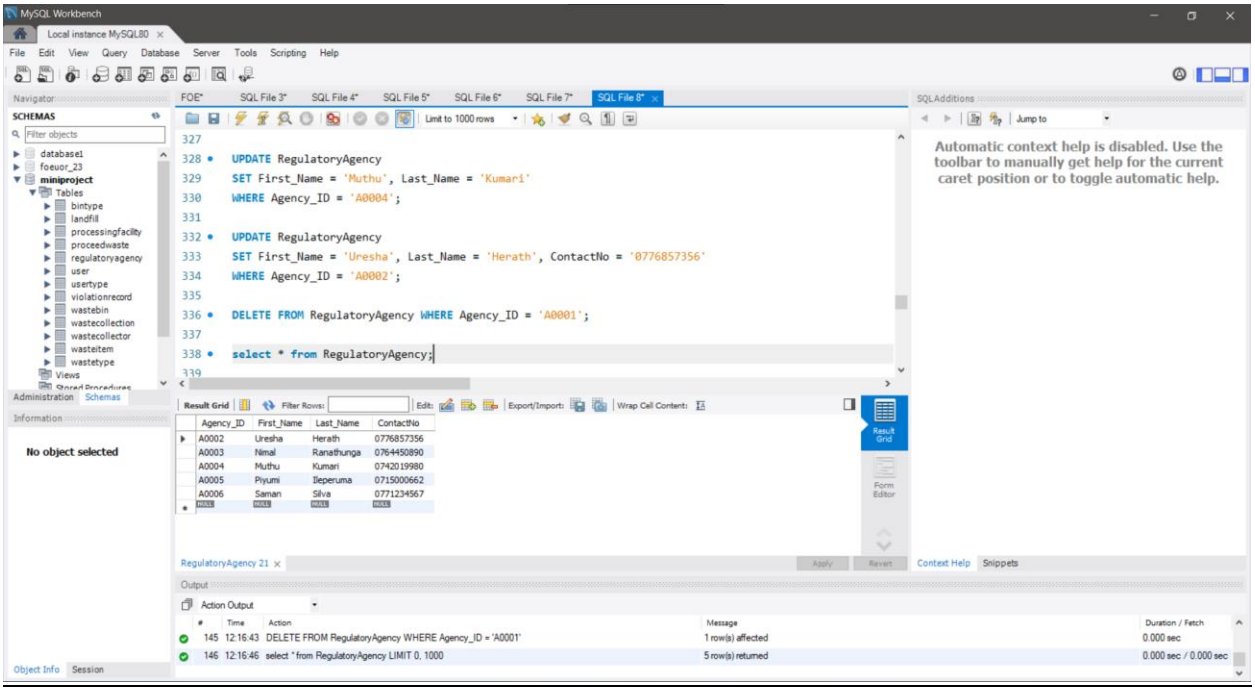
WasteCollector Table



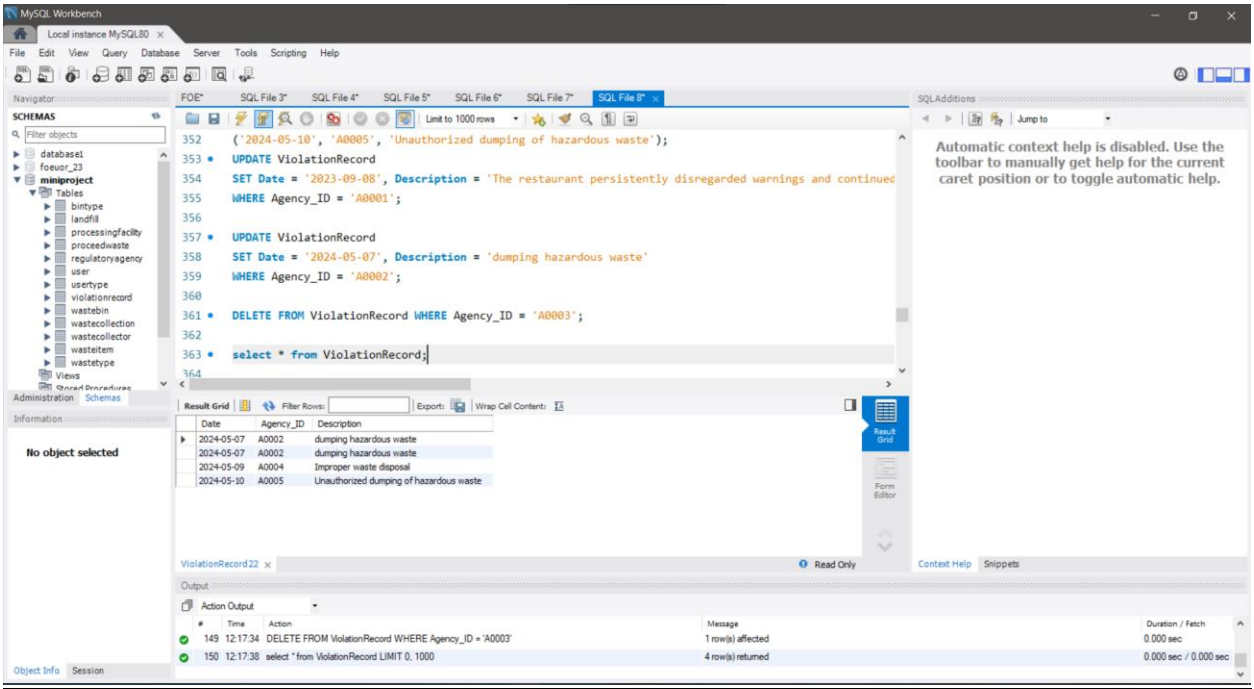
WasteCollection Table



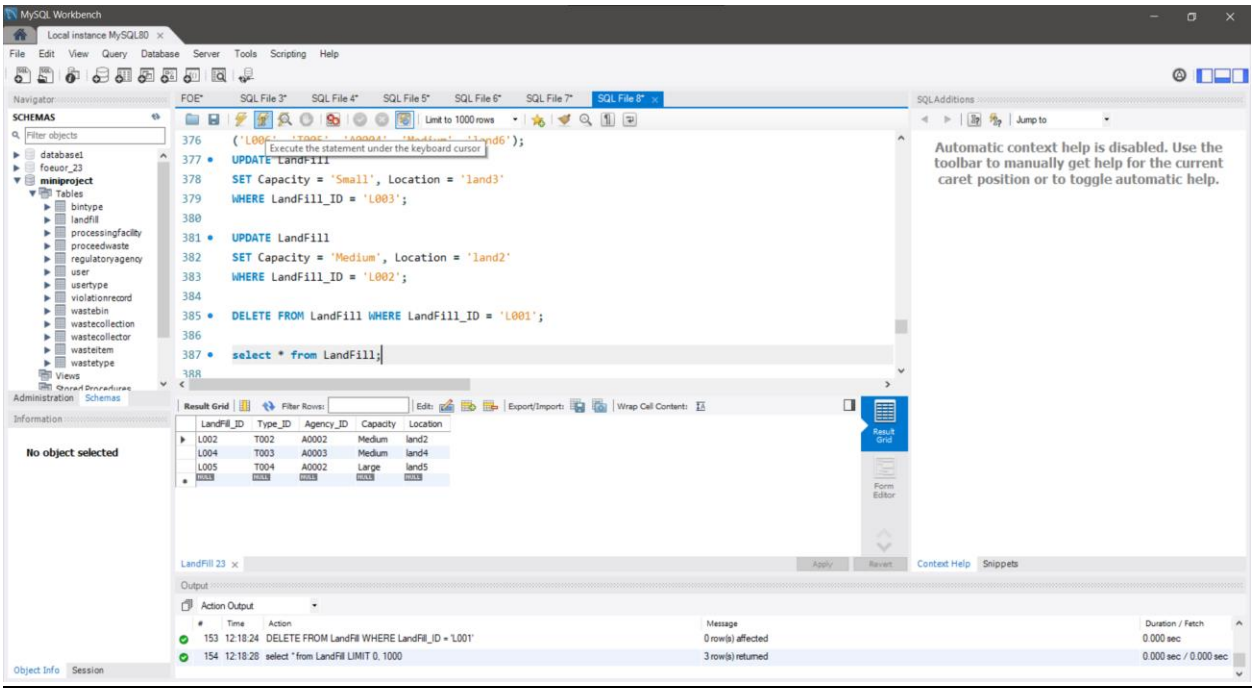
RegulatoryAgency Table



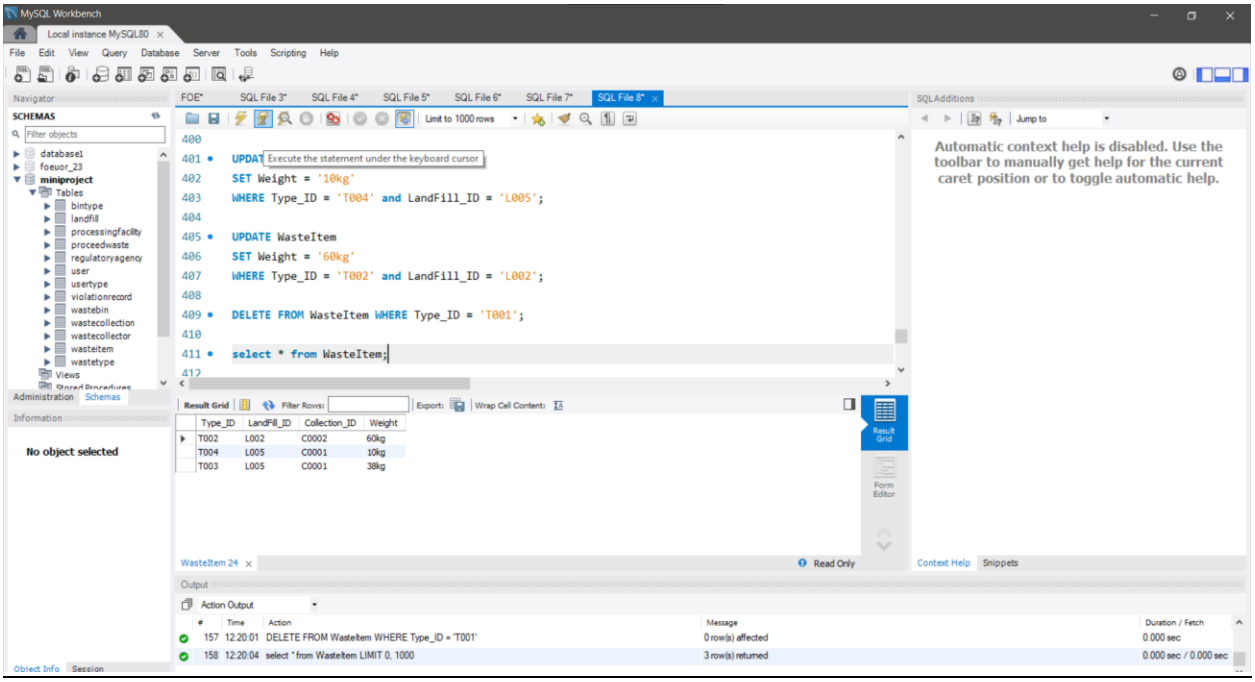
ViolationRecord Table



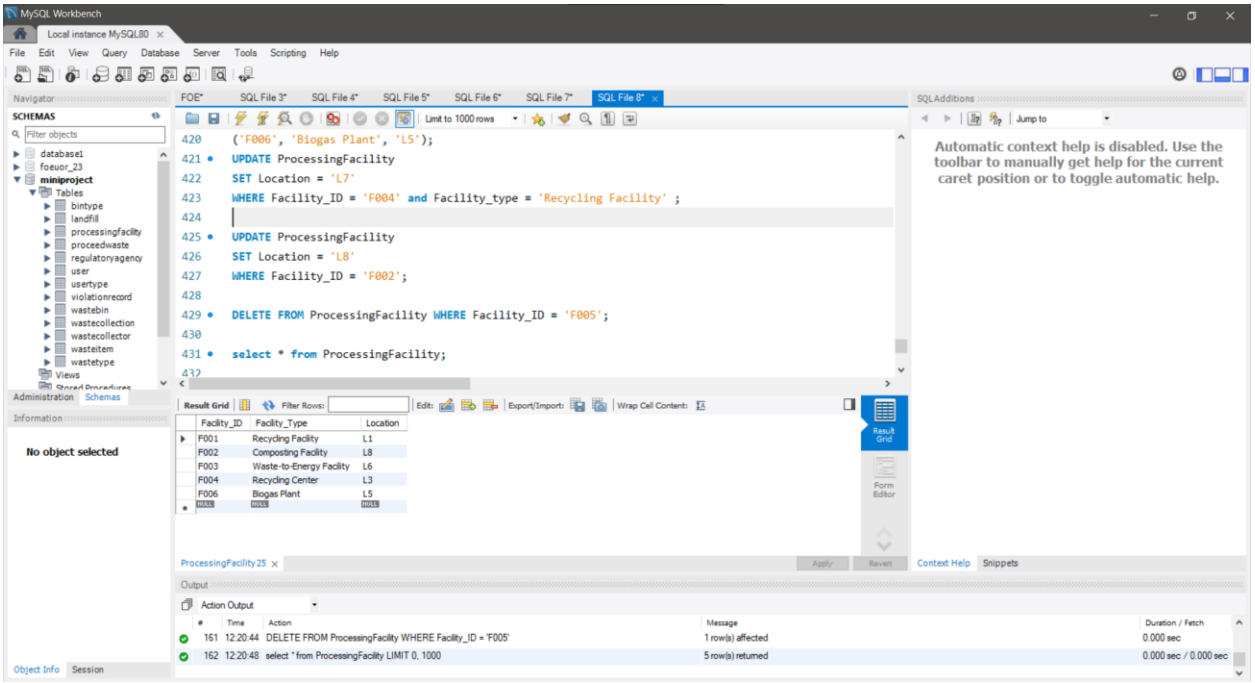
LandFill Table



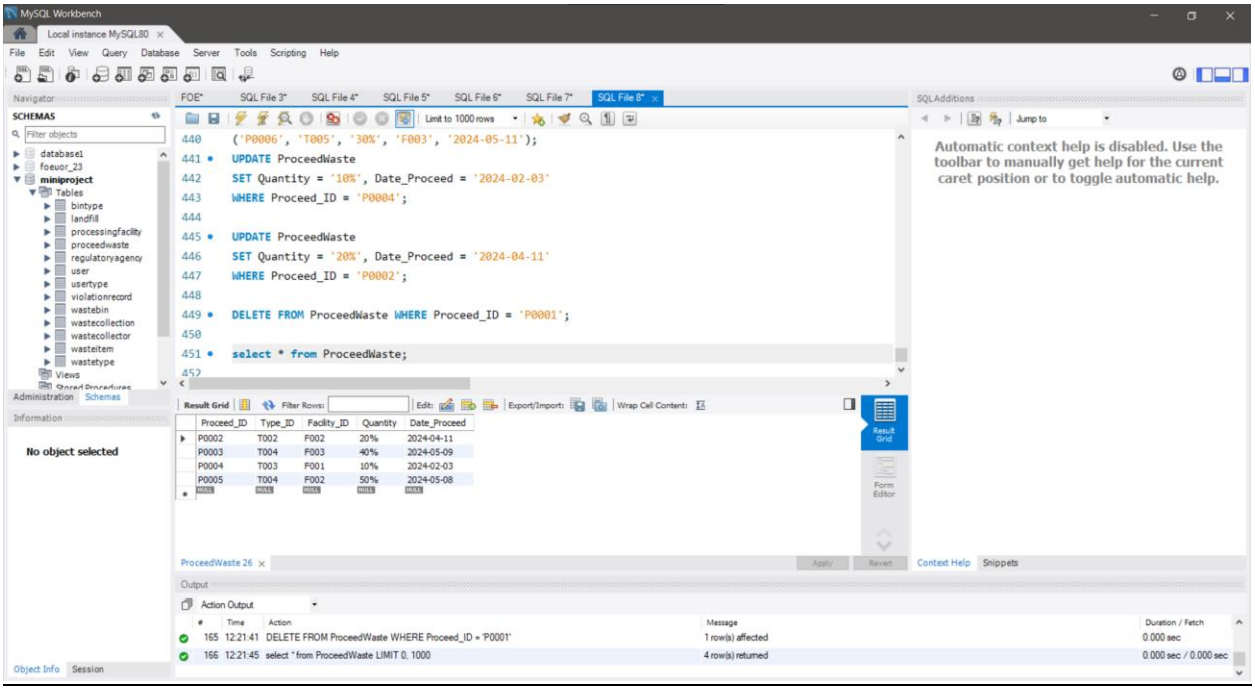
WasteItem Table



ProcessingFacility Table



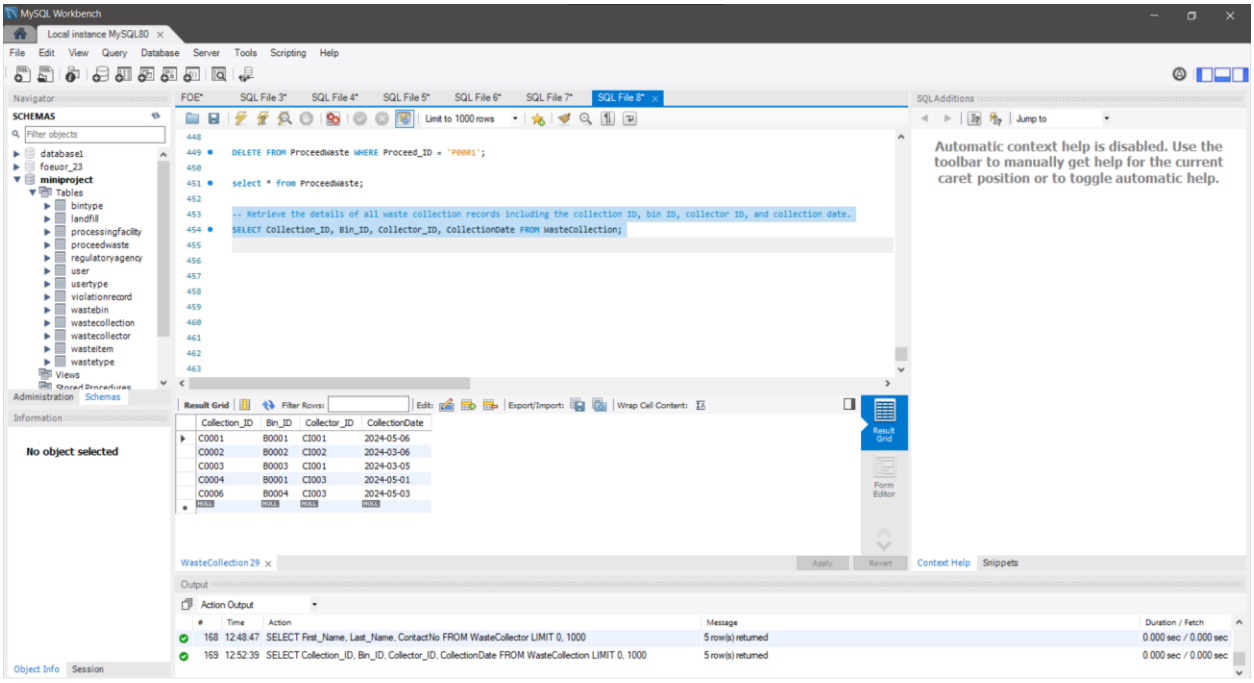
ProceedWaste Table



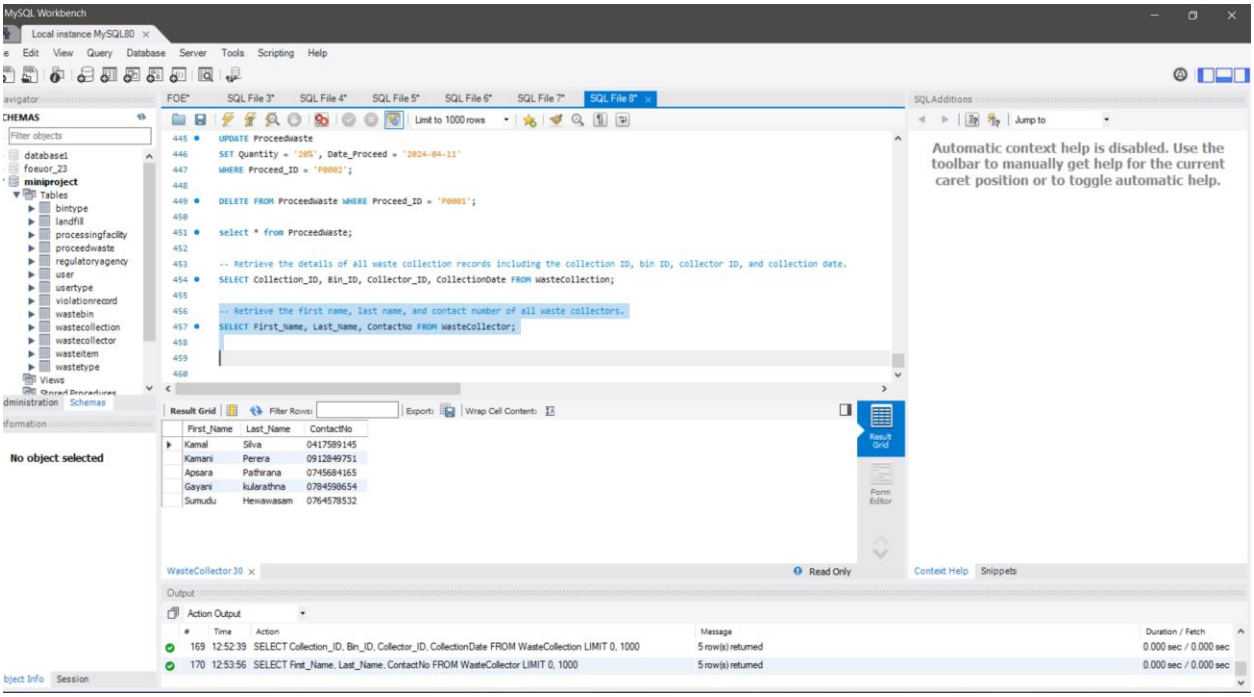
4 Chapter 4- Transactions

Simple Queries

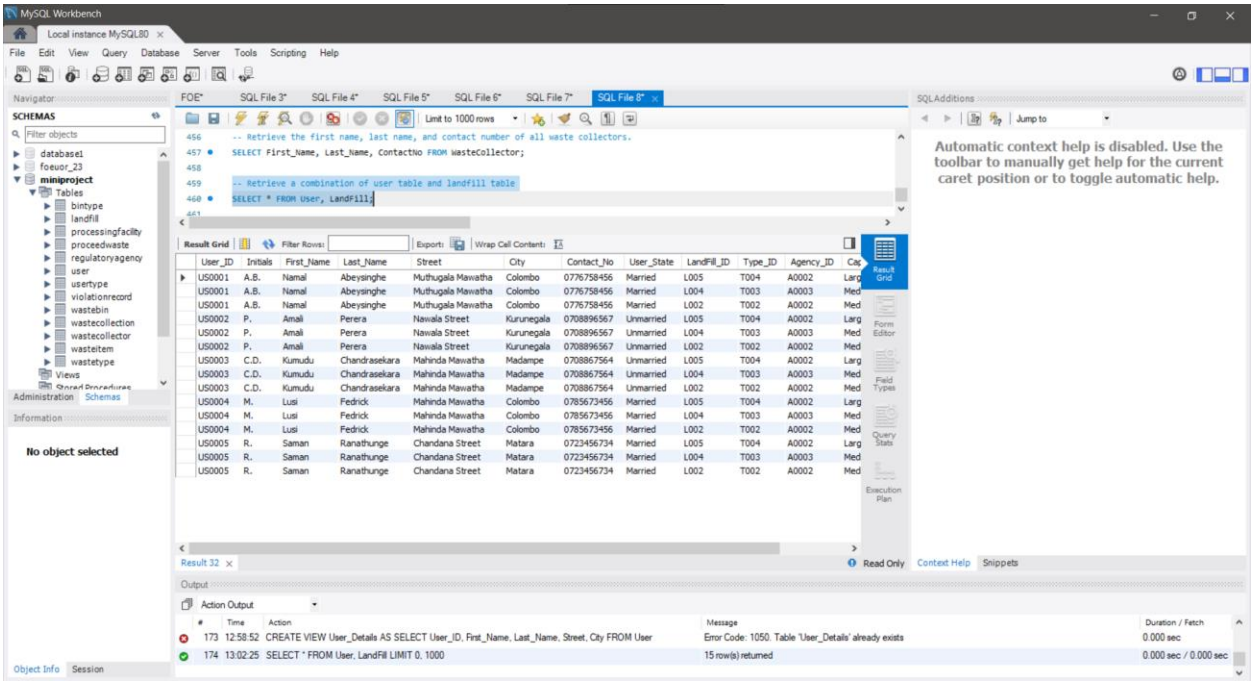
Select Operation



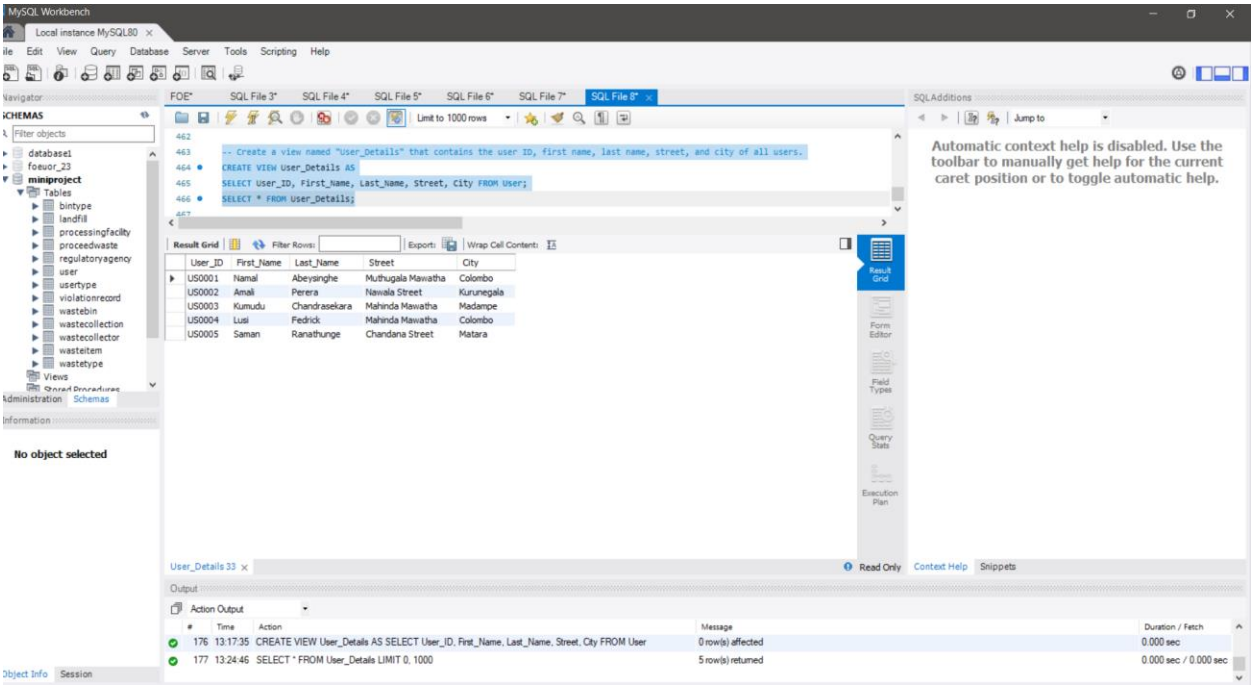
Project Operation



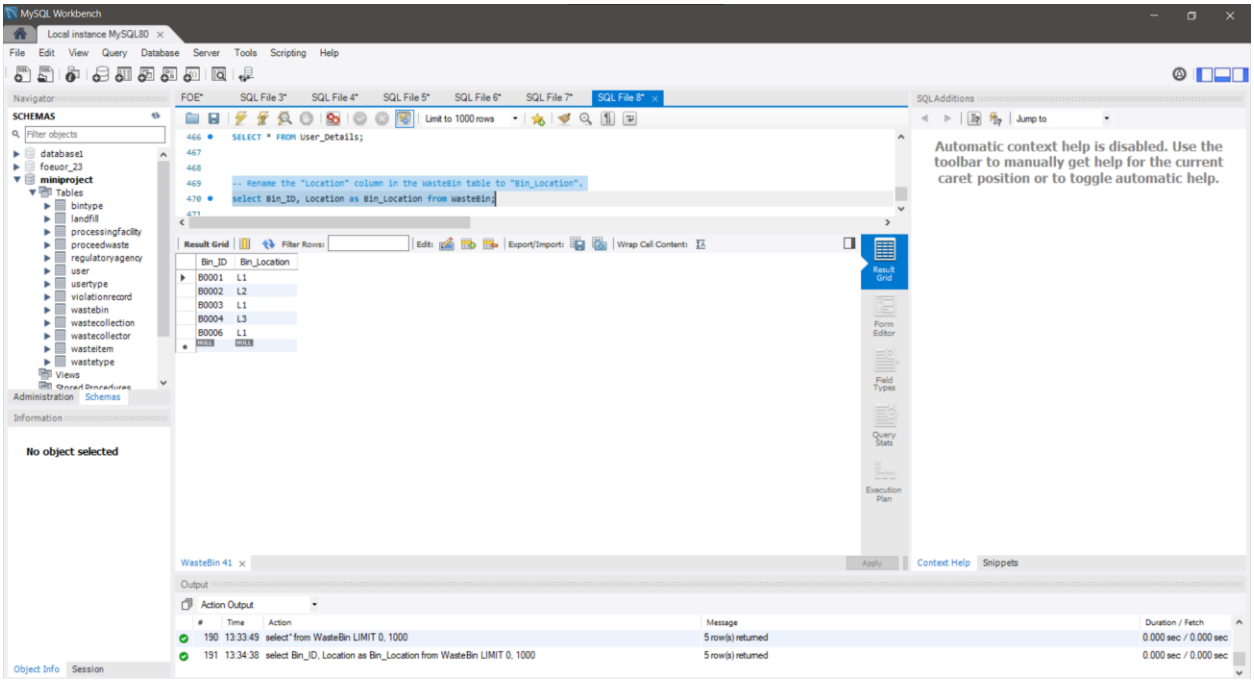
Cartesian Product Operation



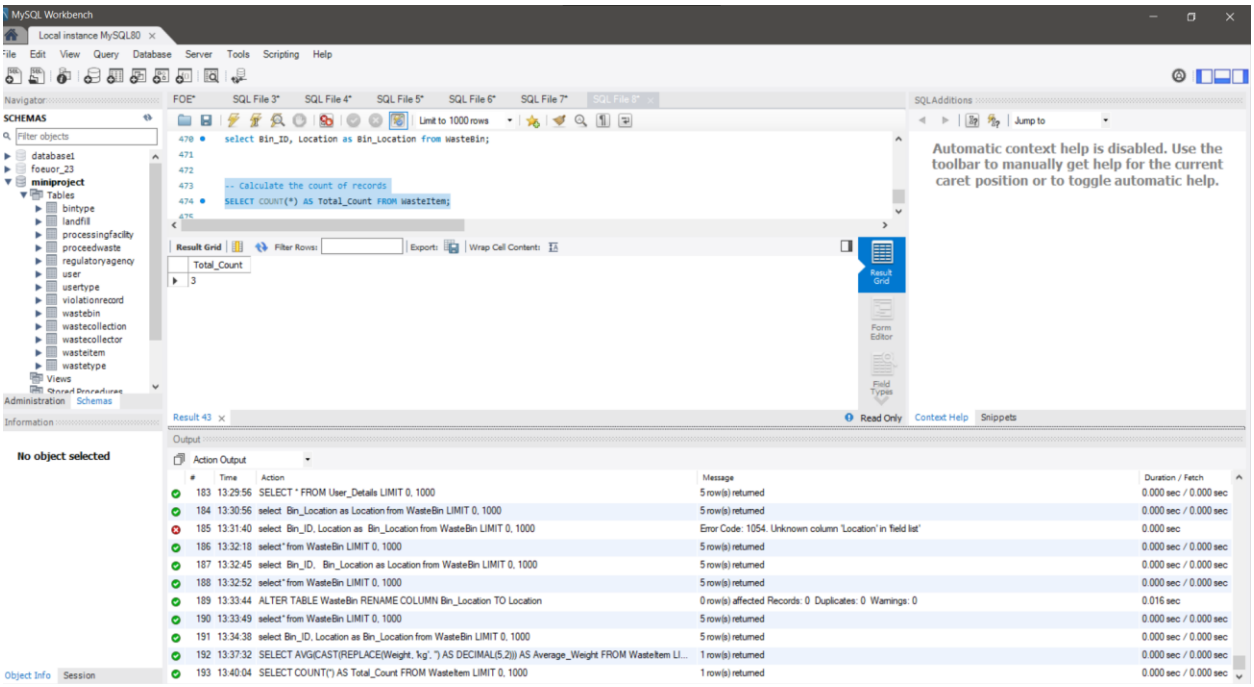
Creating a User View



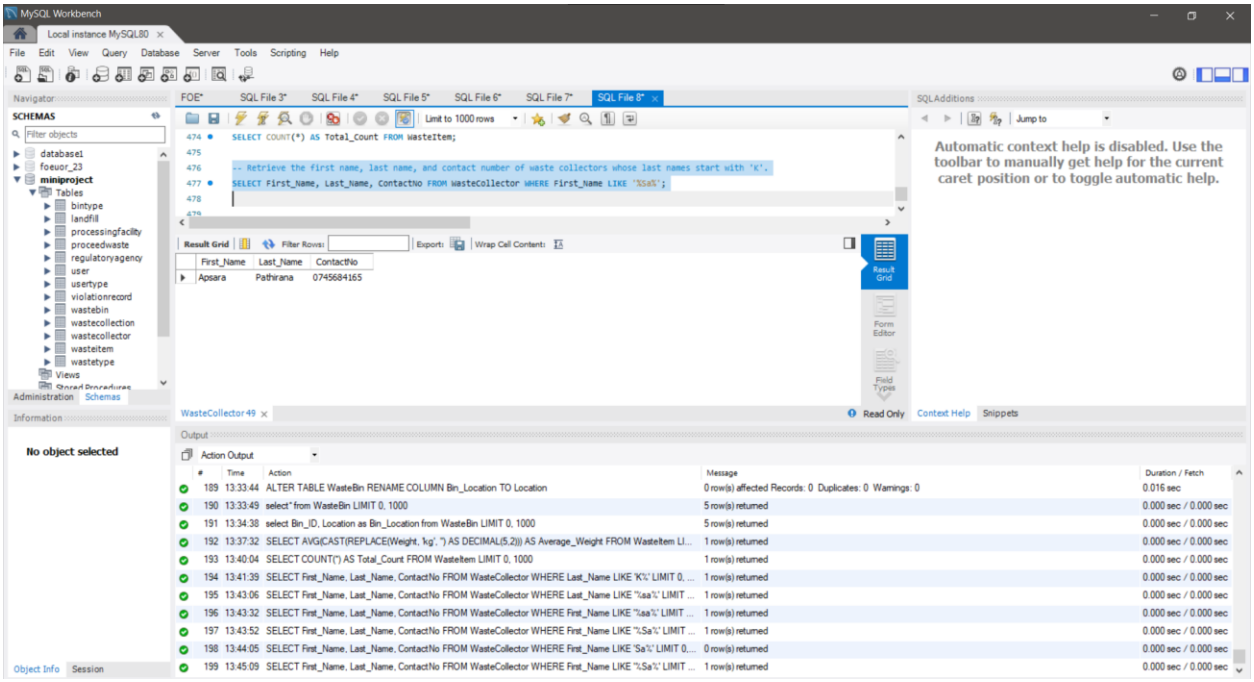
Renaming Operation



Aggregation Function

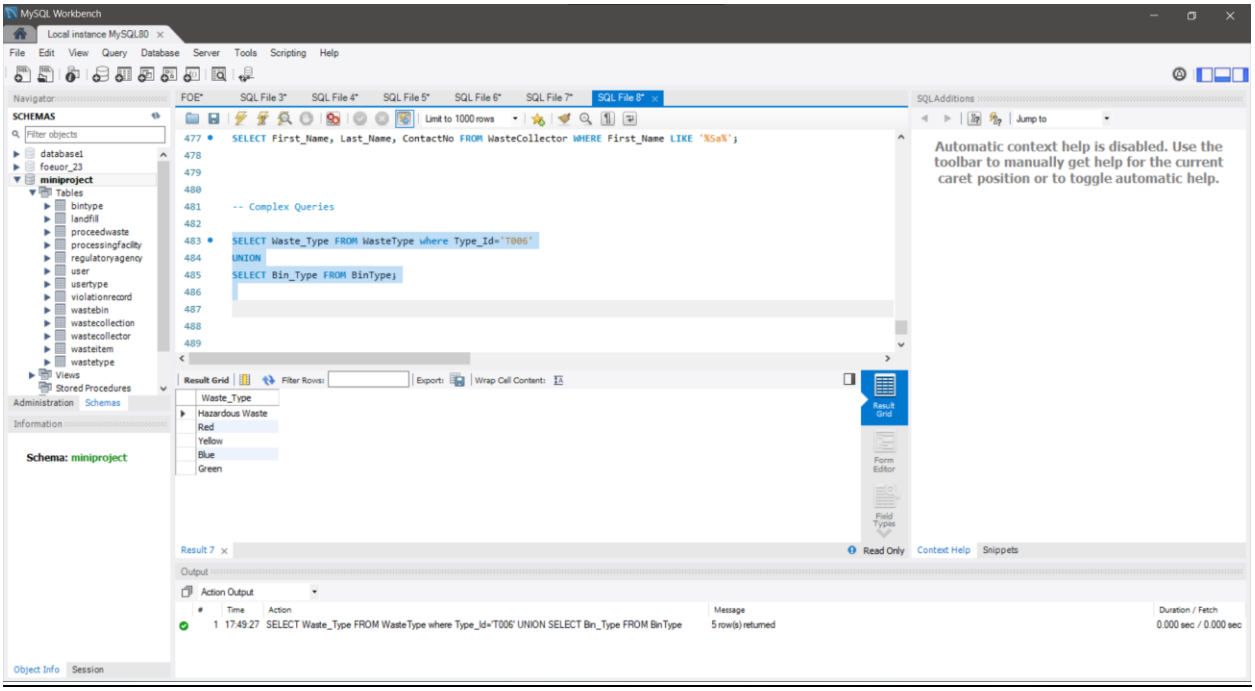


Use Of LIKE keyword

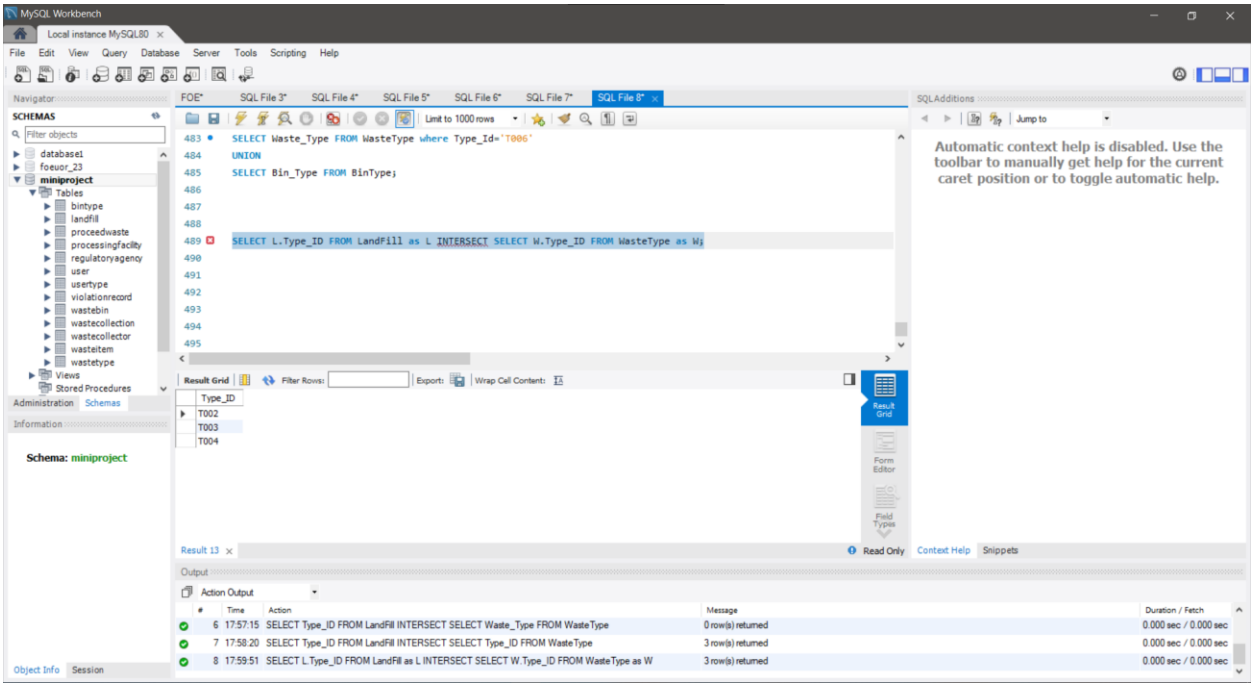


Complex Queries

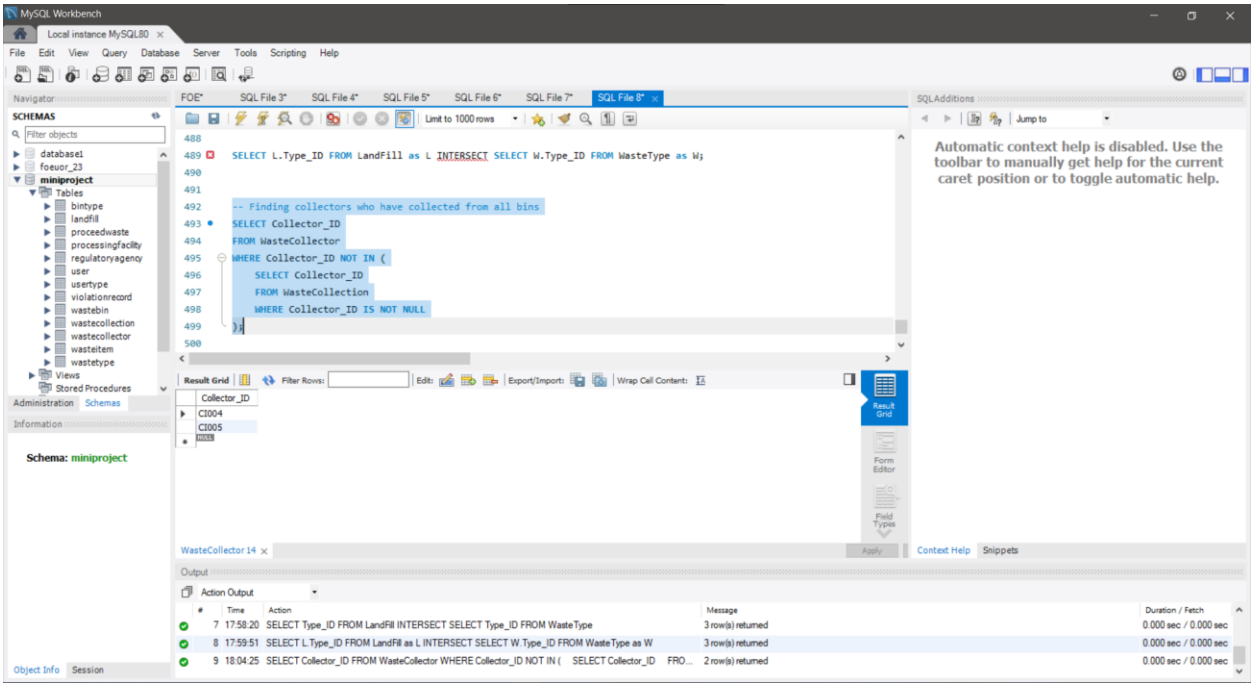
Union



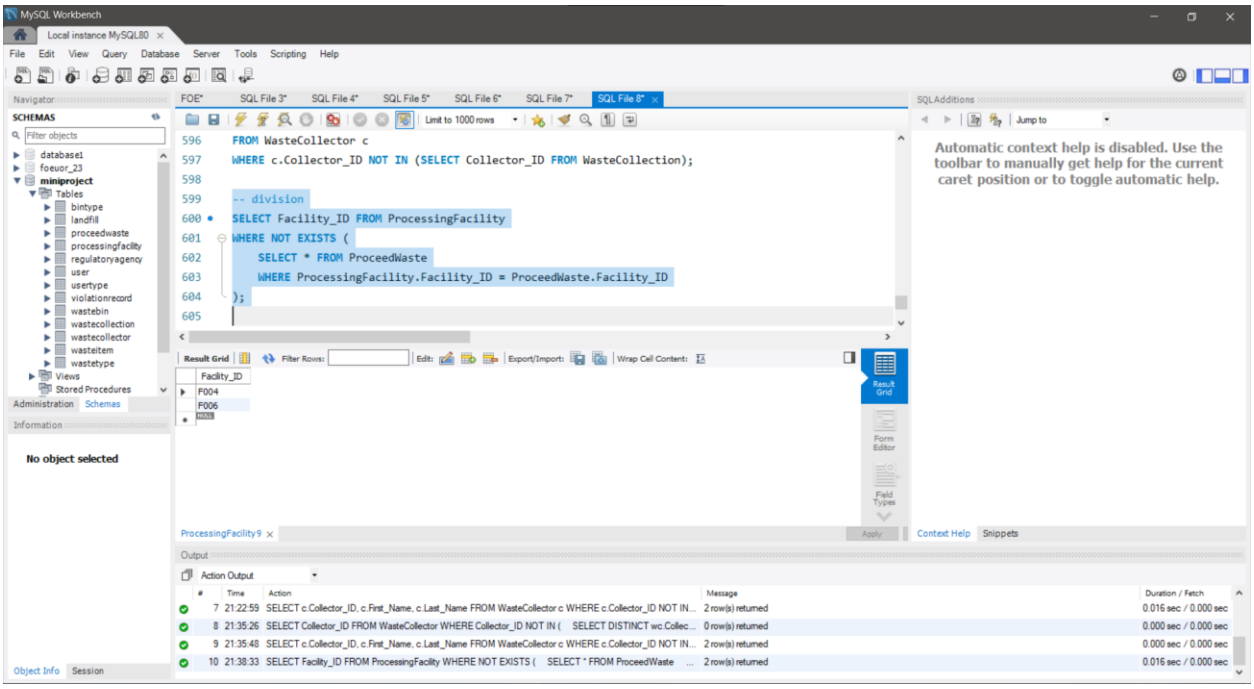
Intersection



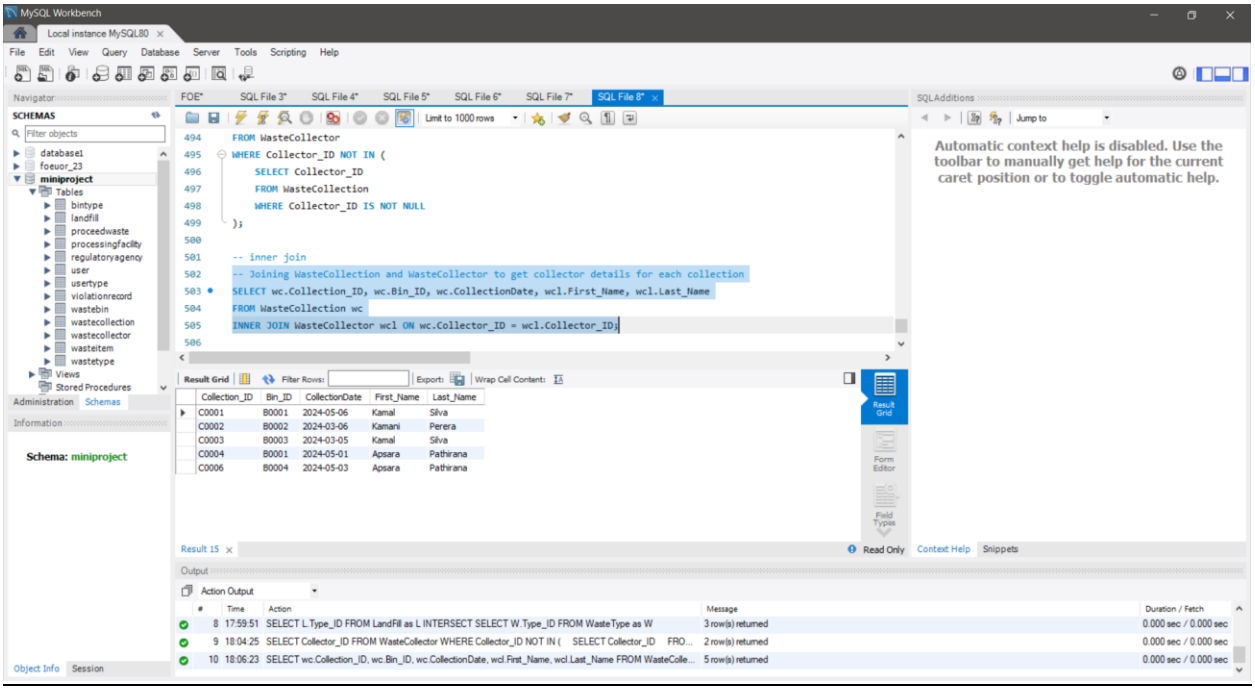
Set Difference



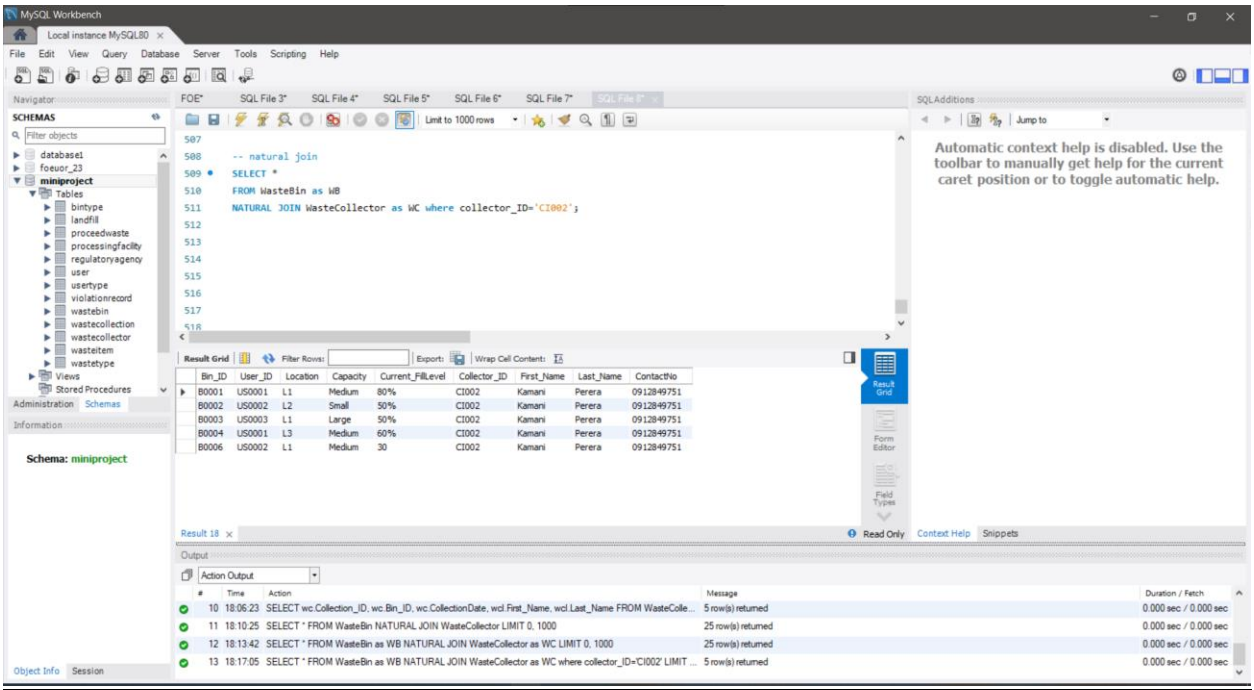
Division



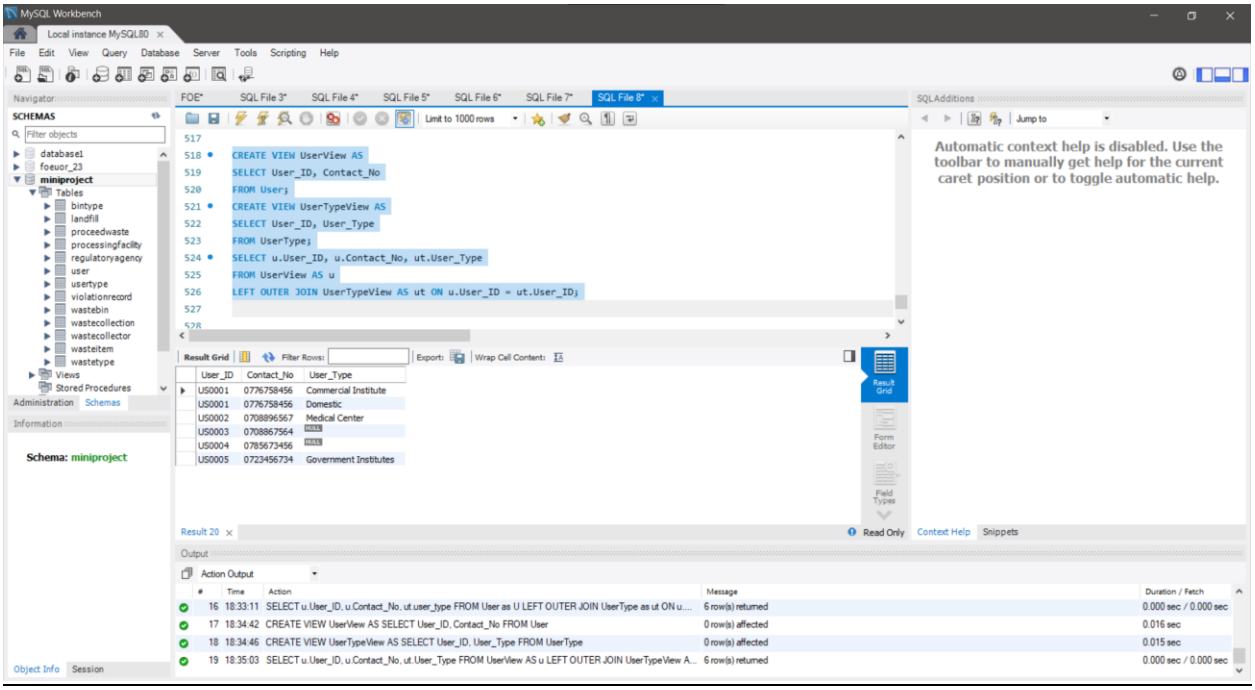
Inner join using User Views



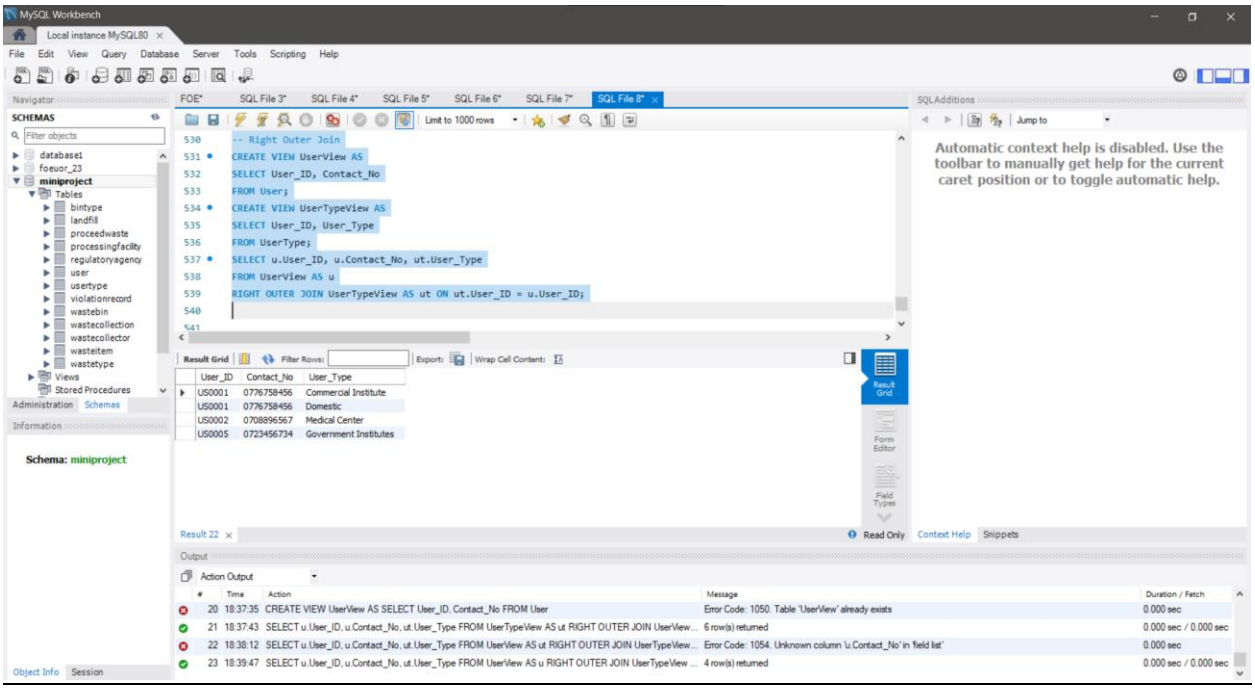
Natural Join using User Views



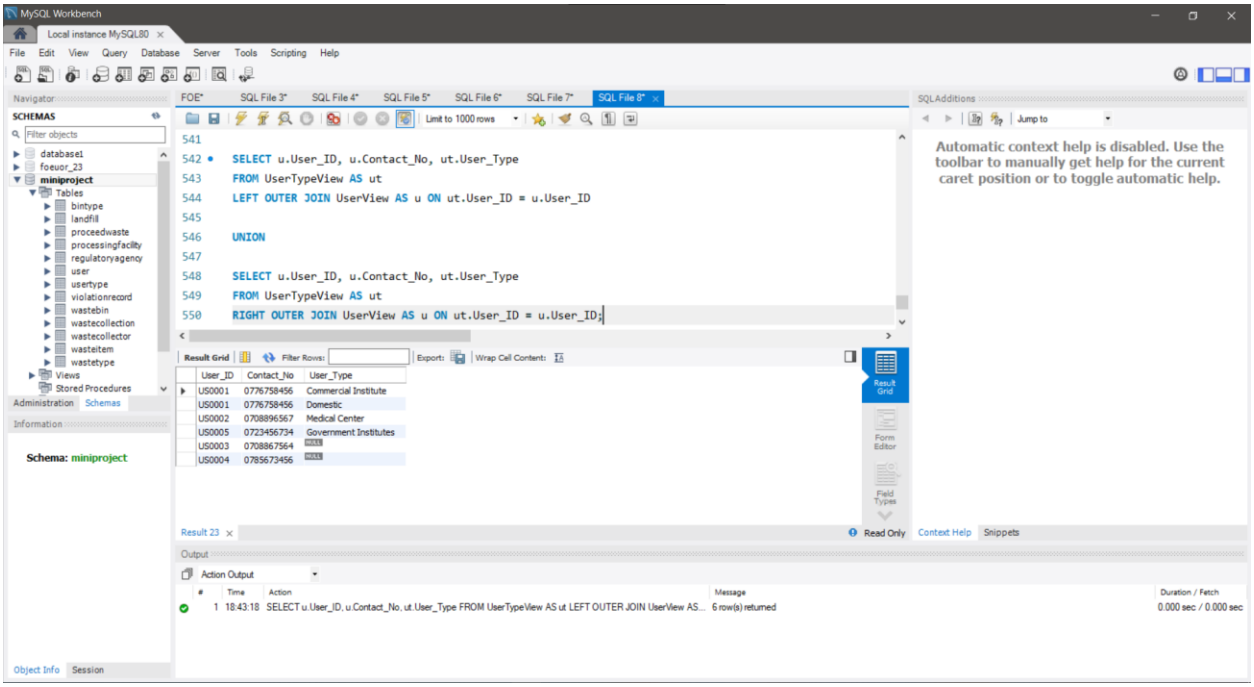
Left Outer Join



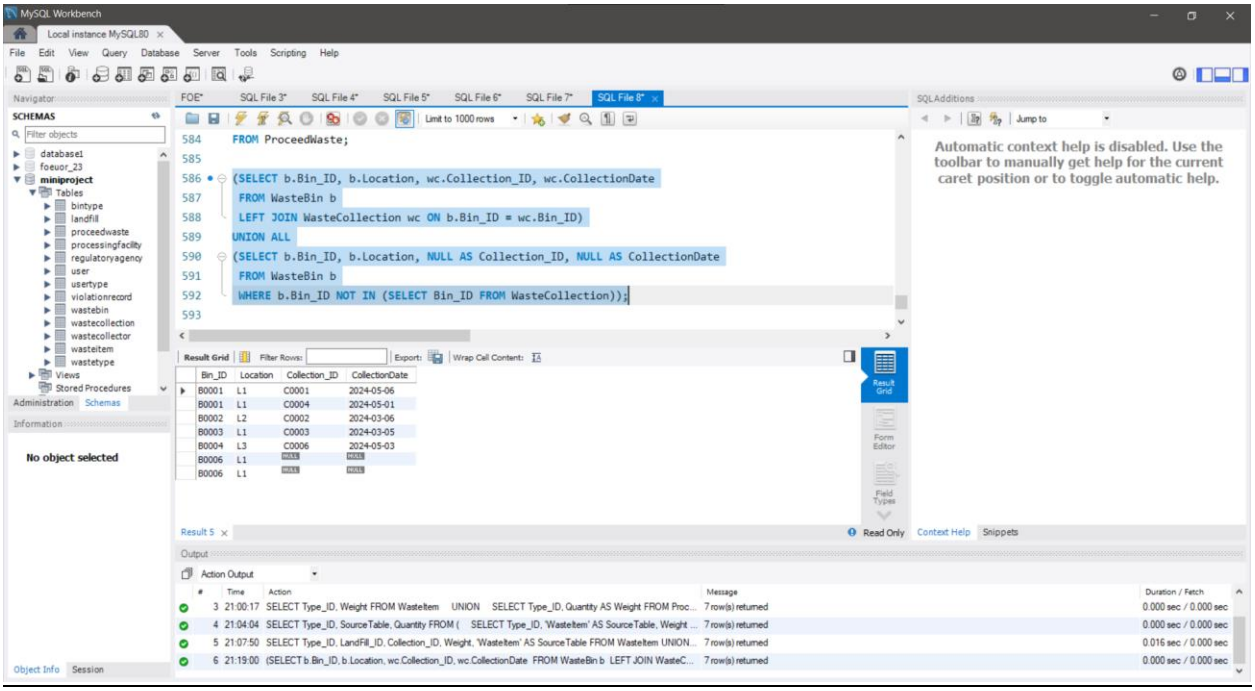
Right Outer Join



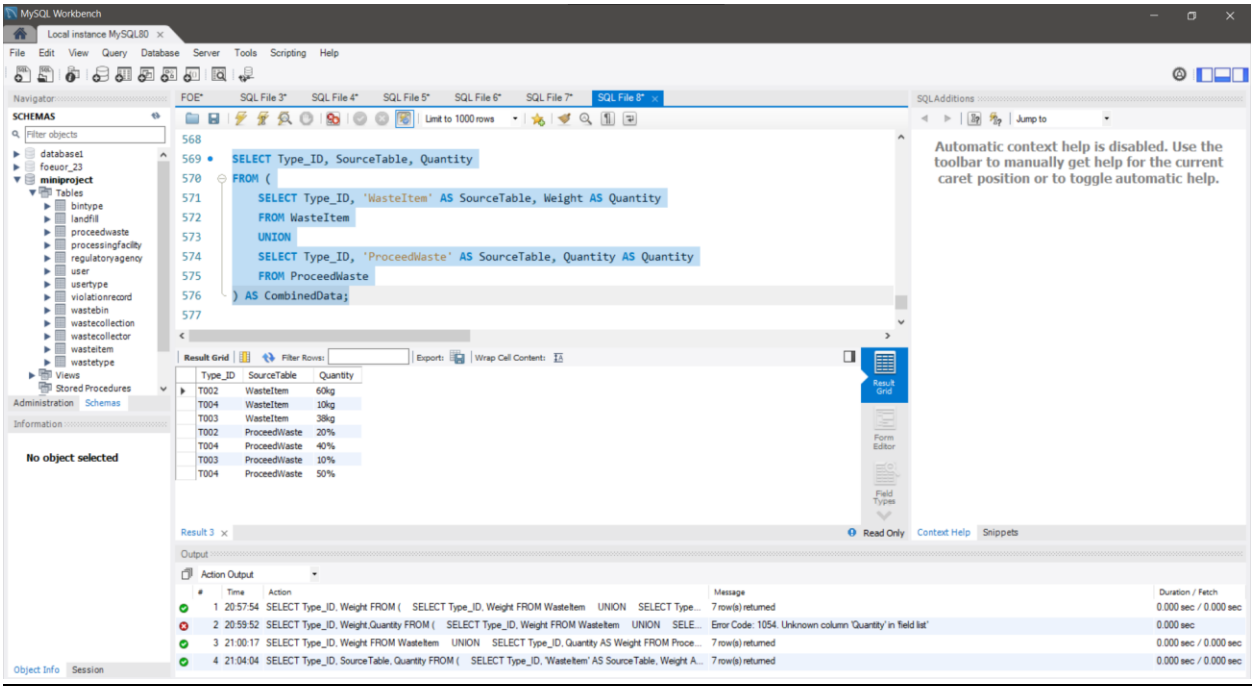
Full Outer Join



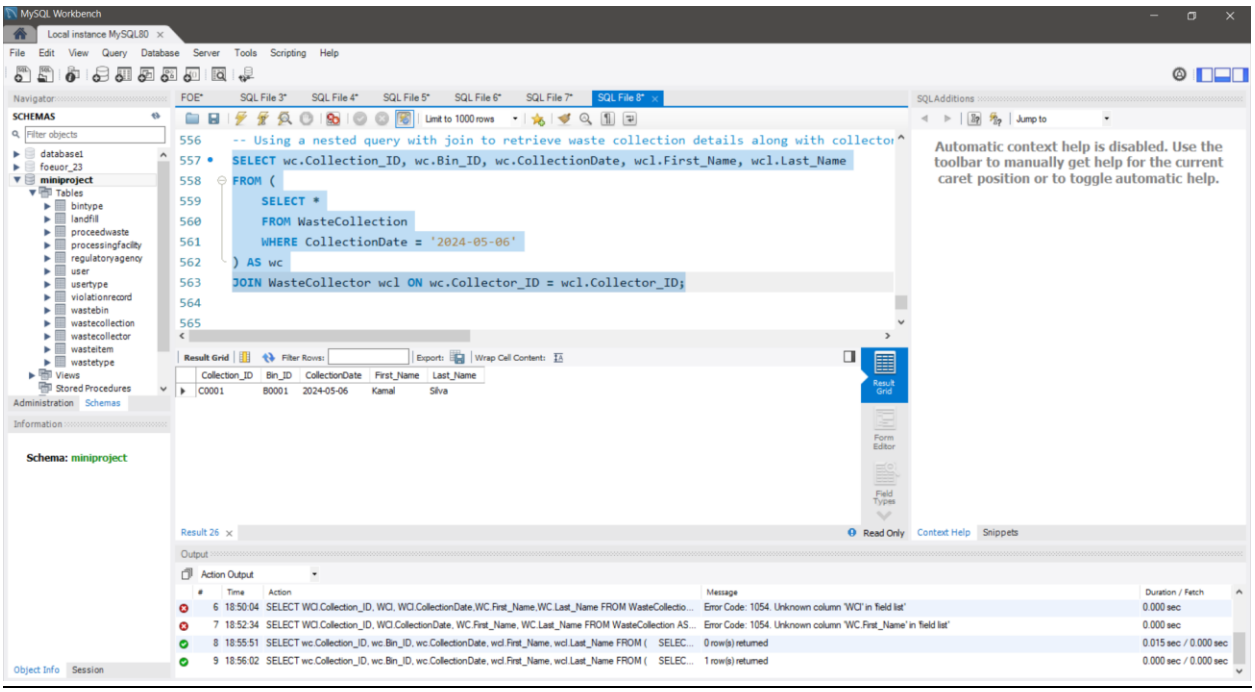
Outer Union Relational Algebraic Operation



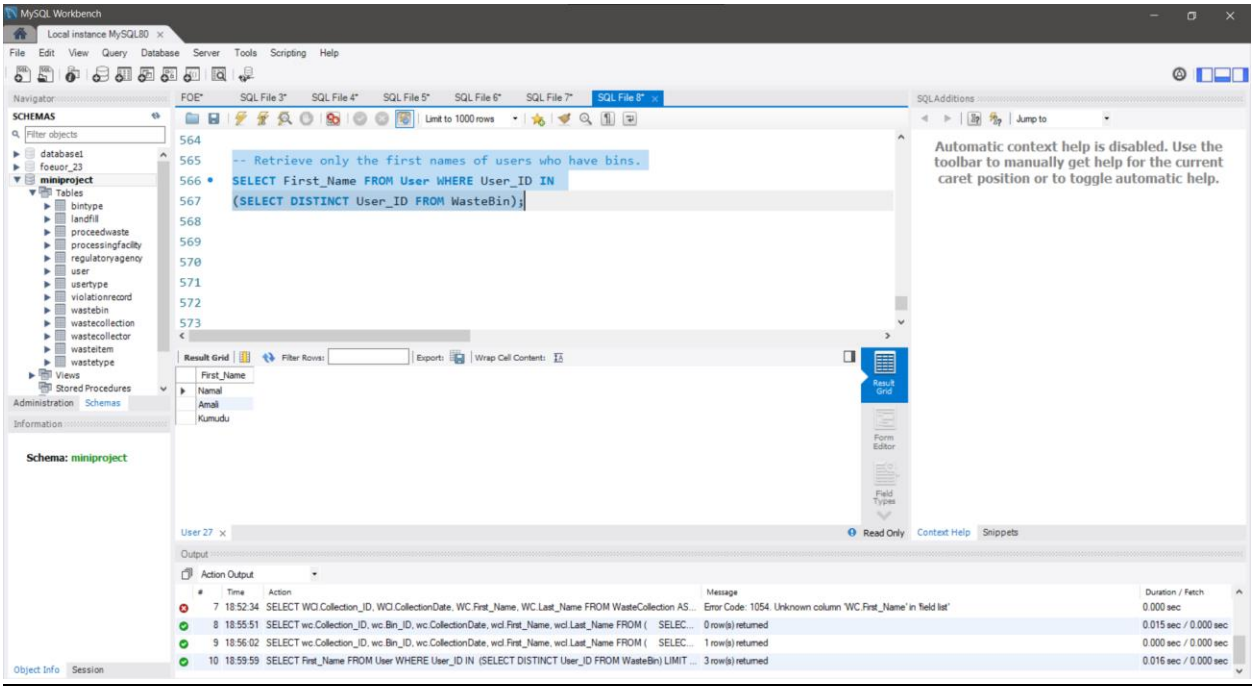
Nested Query with Union



Nested Query With Join



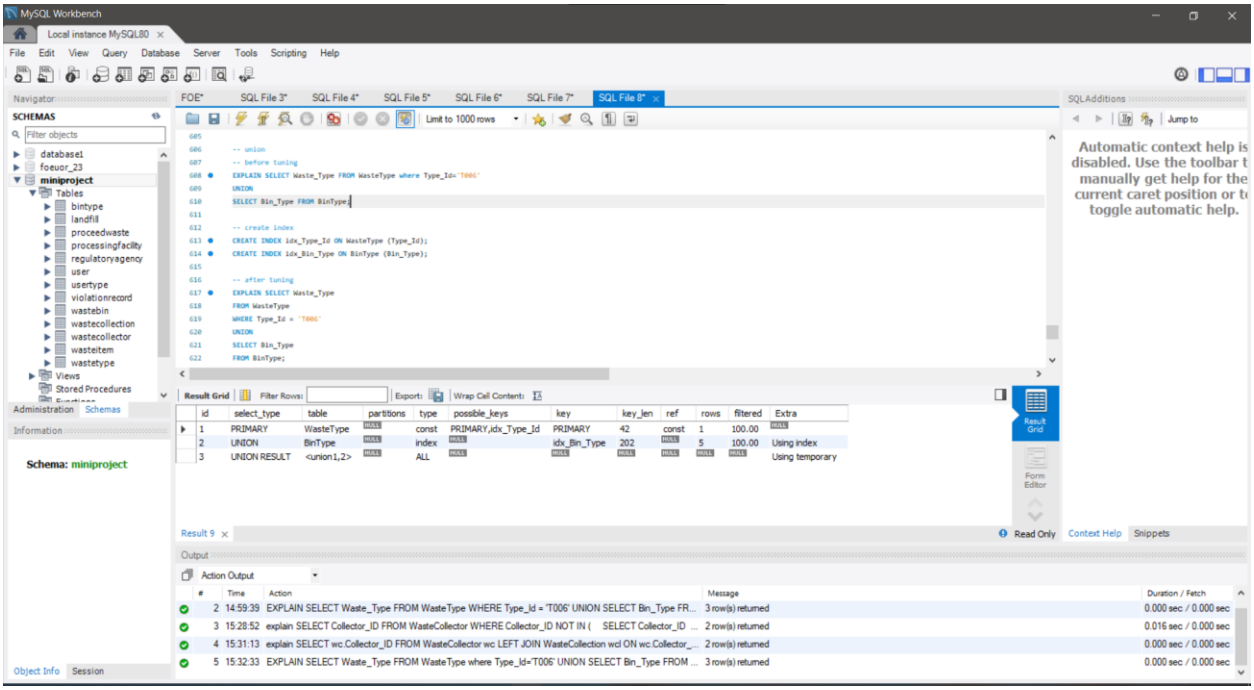
Nested Query with Projection



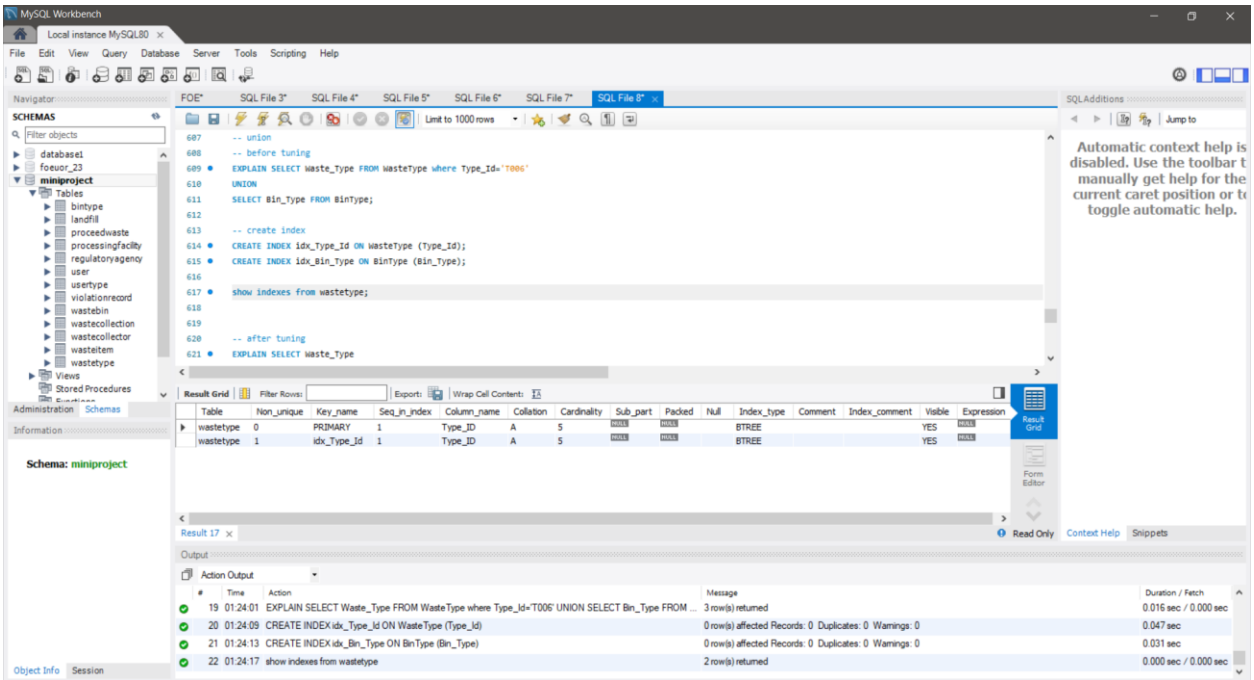
5 Chapter 5- Data Base Tunning

Union

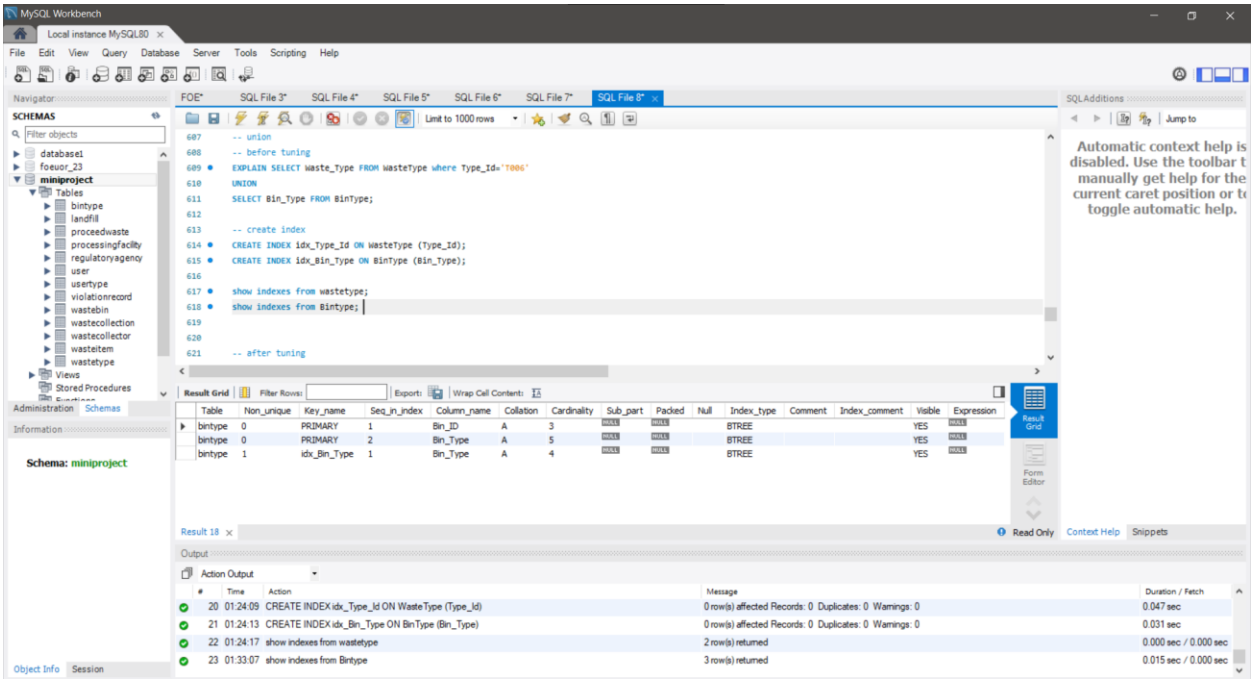
Before tuning



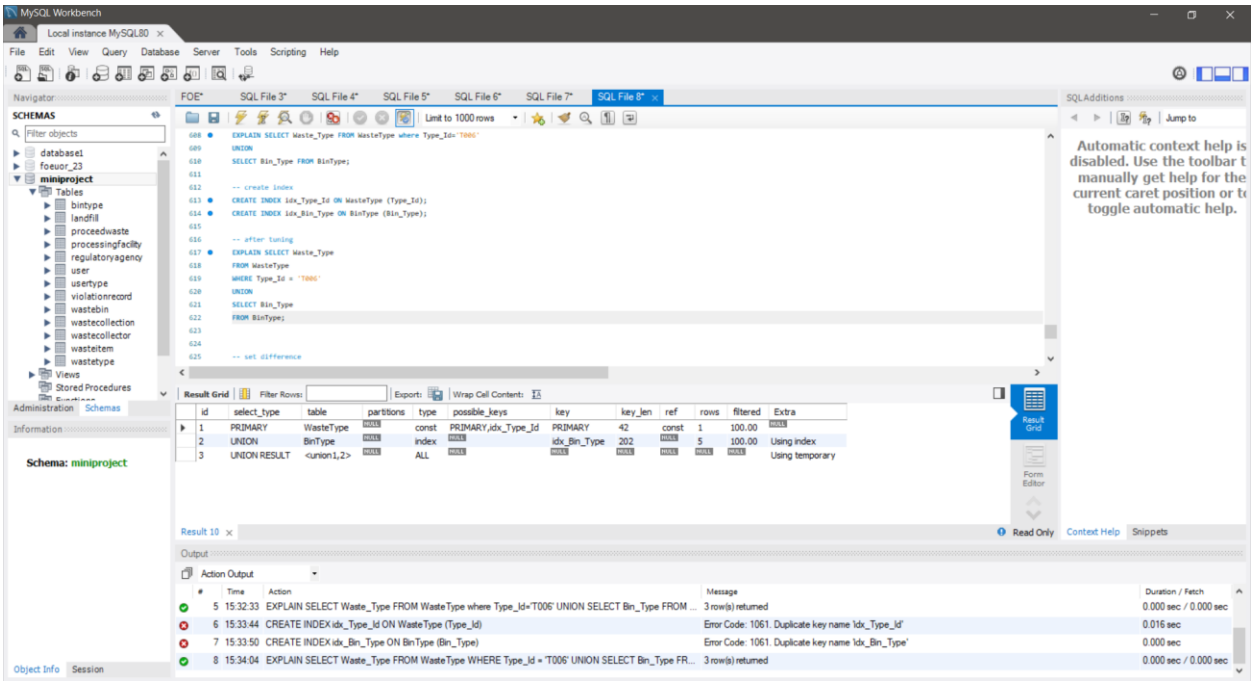
Create index WasteType



Create index in BinType

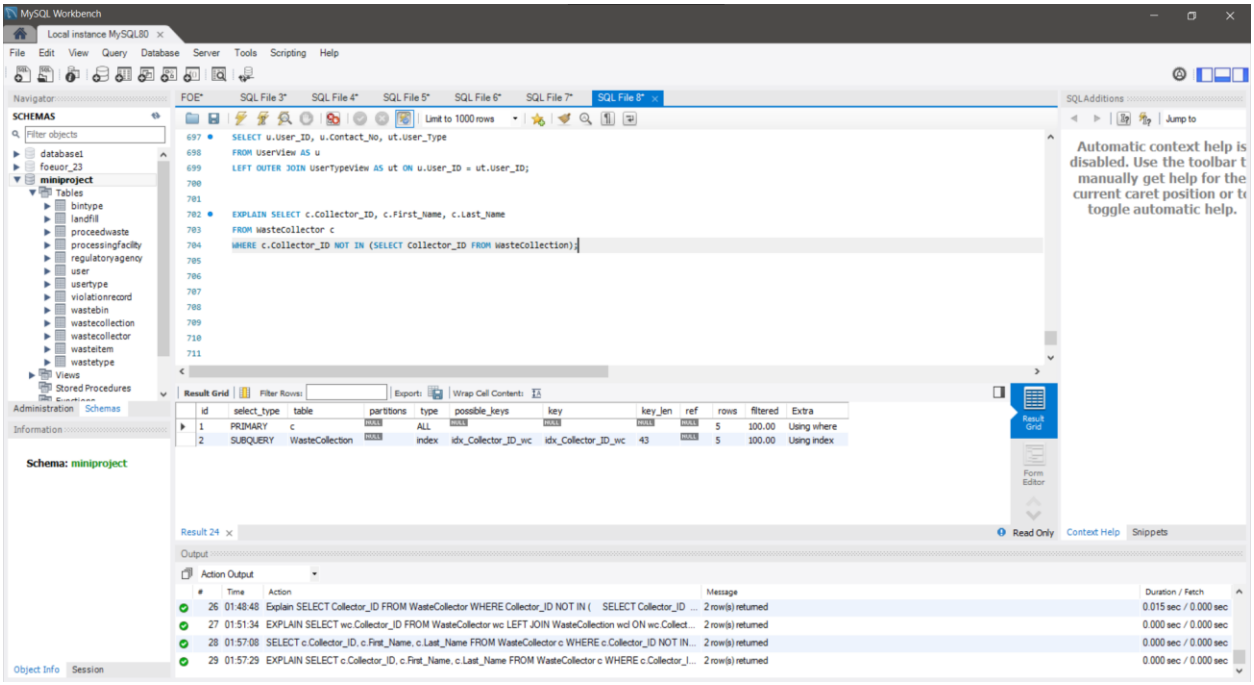


After Tuning

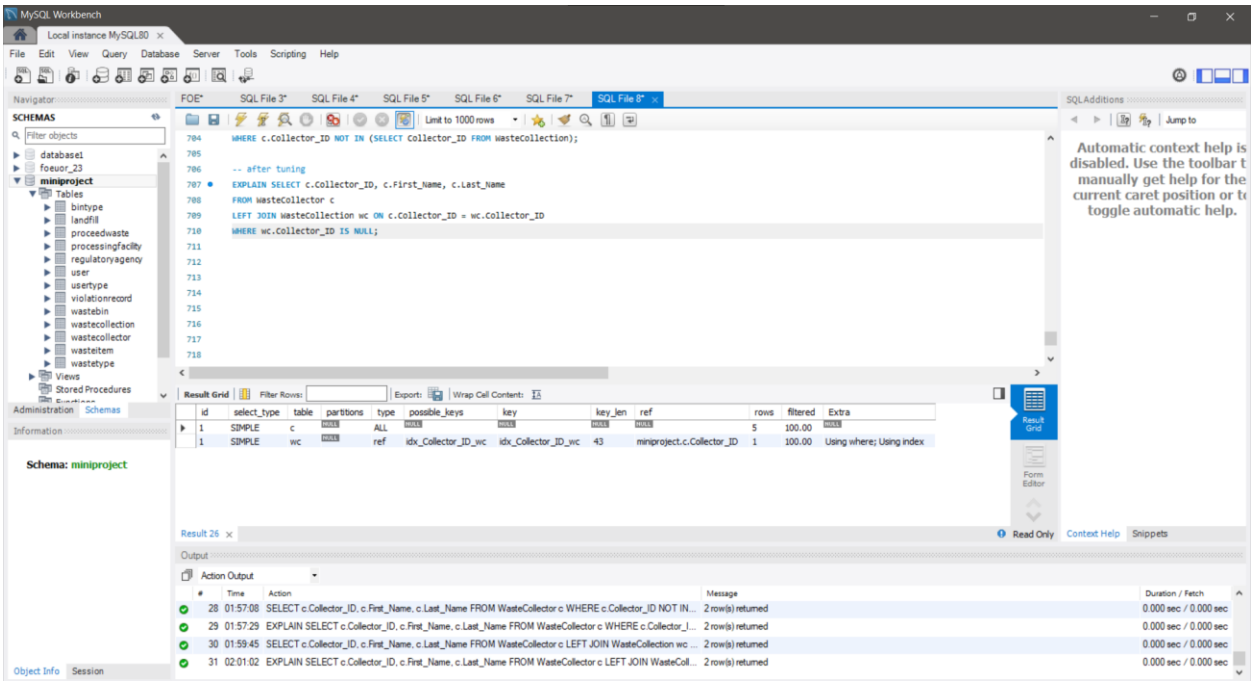


Set Difference

Before Tuning

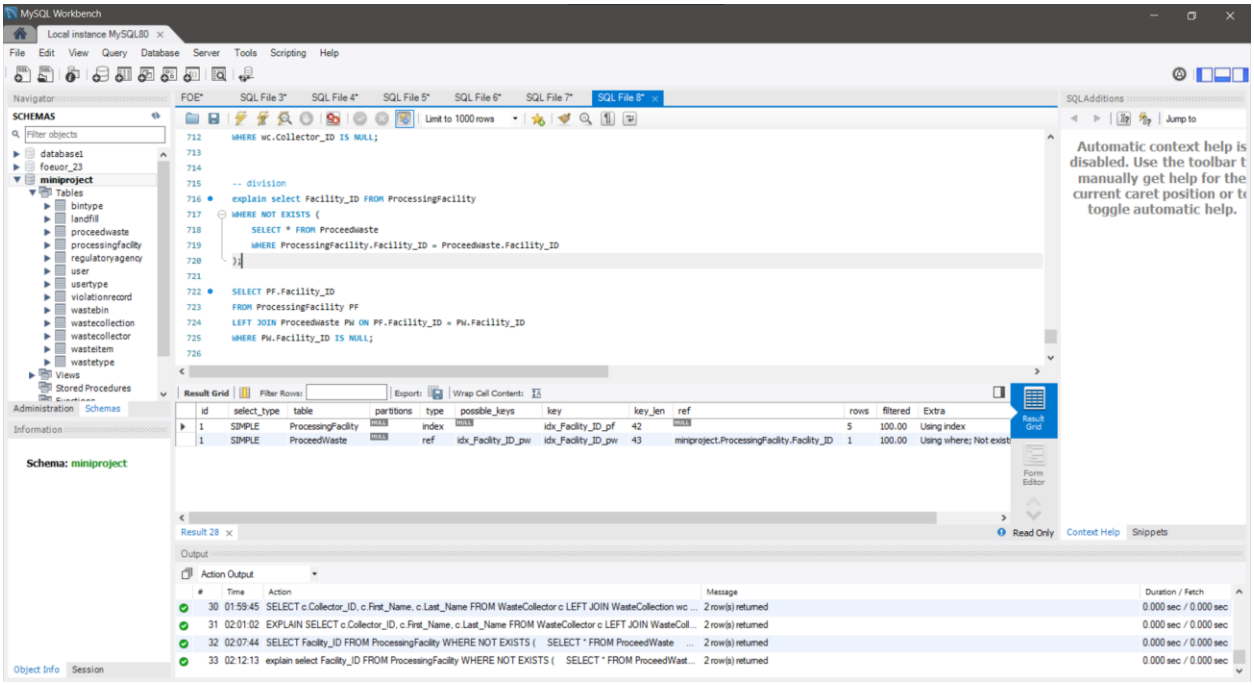


After tuning

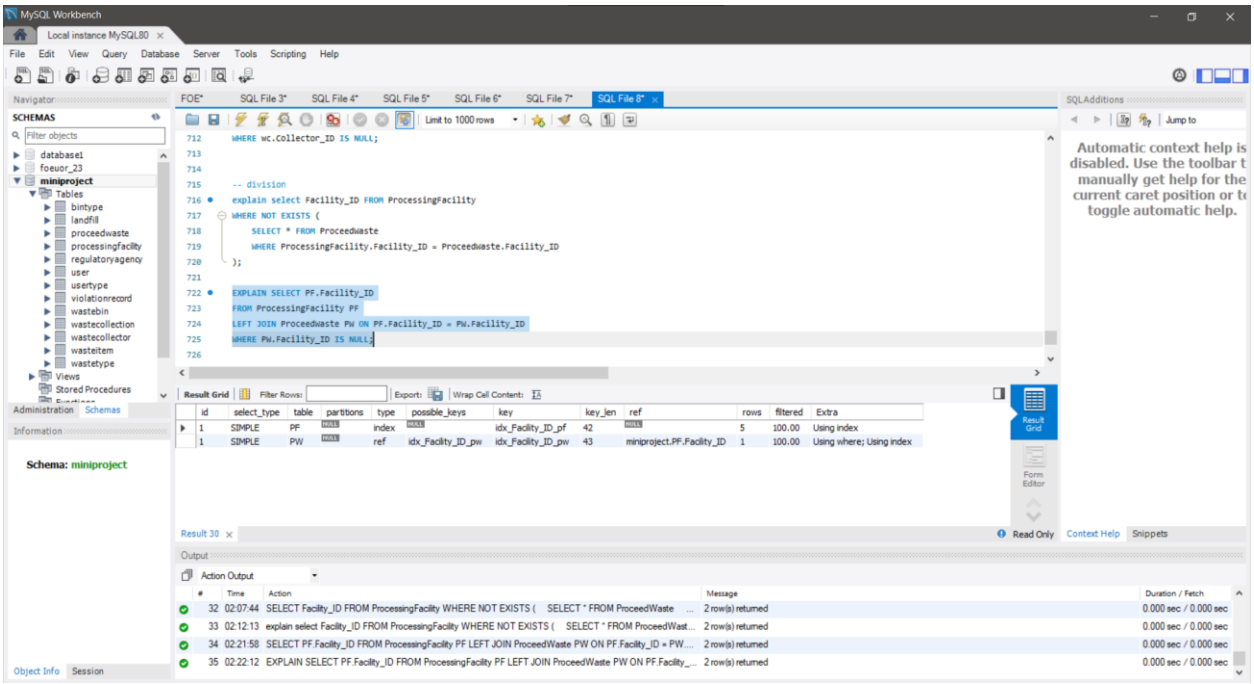


Division

Before Tunning

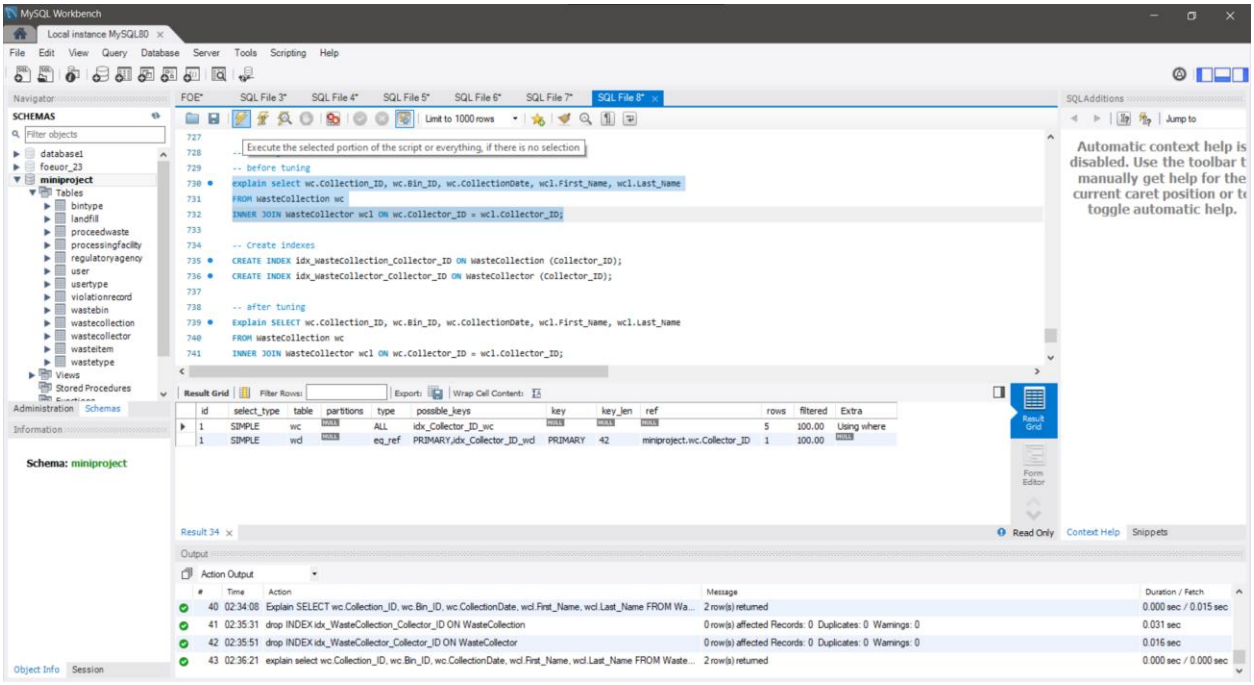


After Tunning

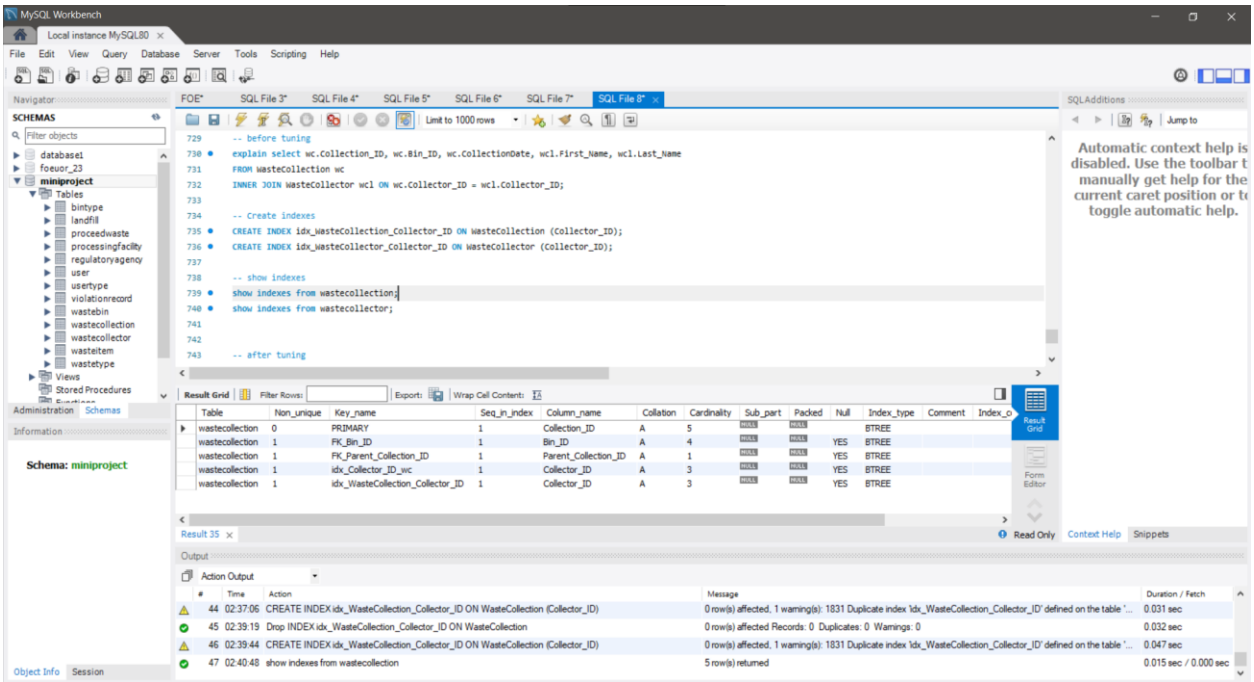


Inner Join

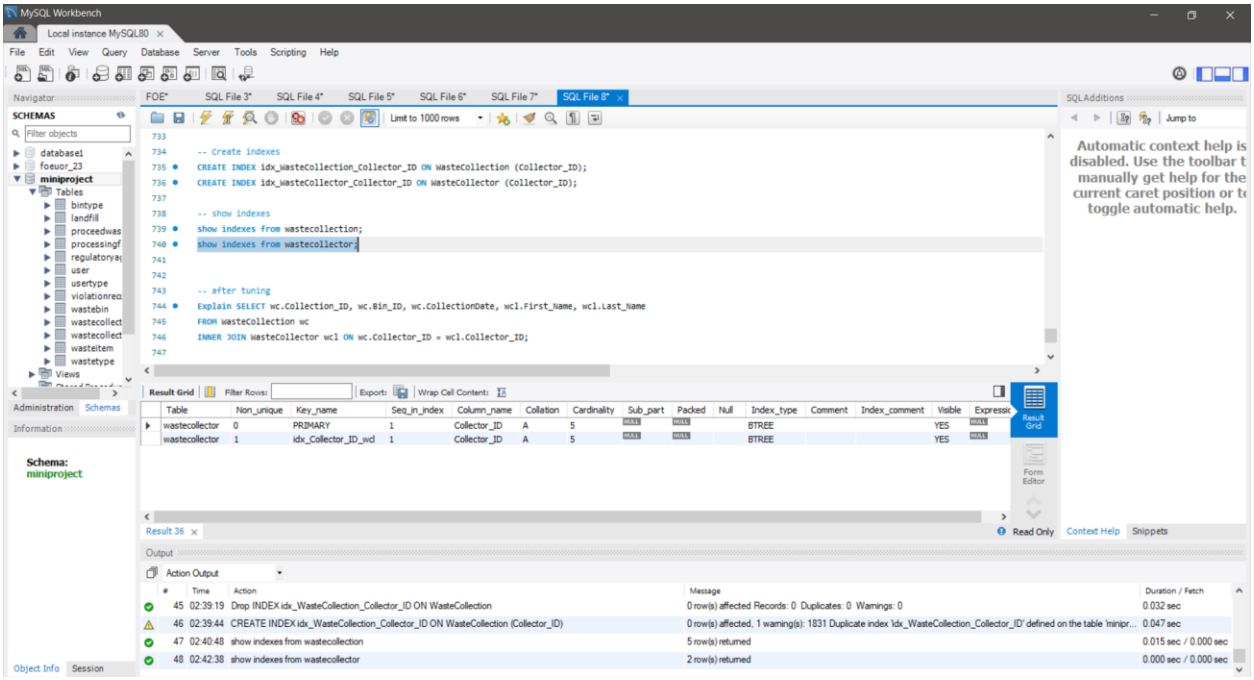
Before Tunning



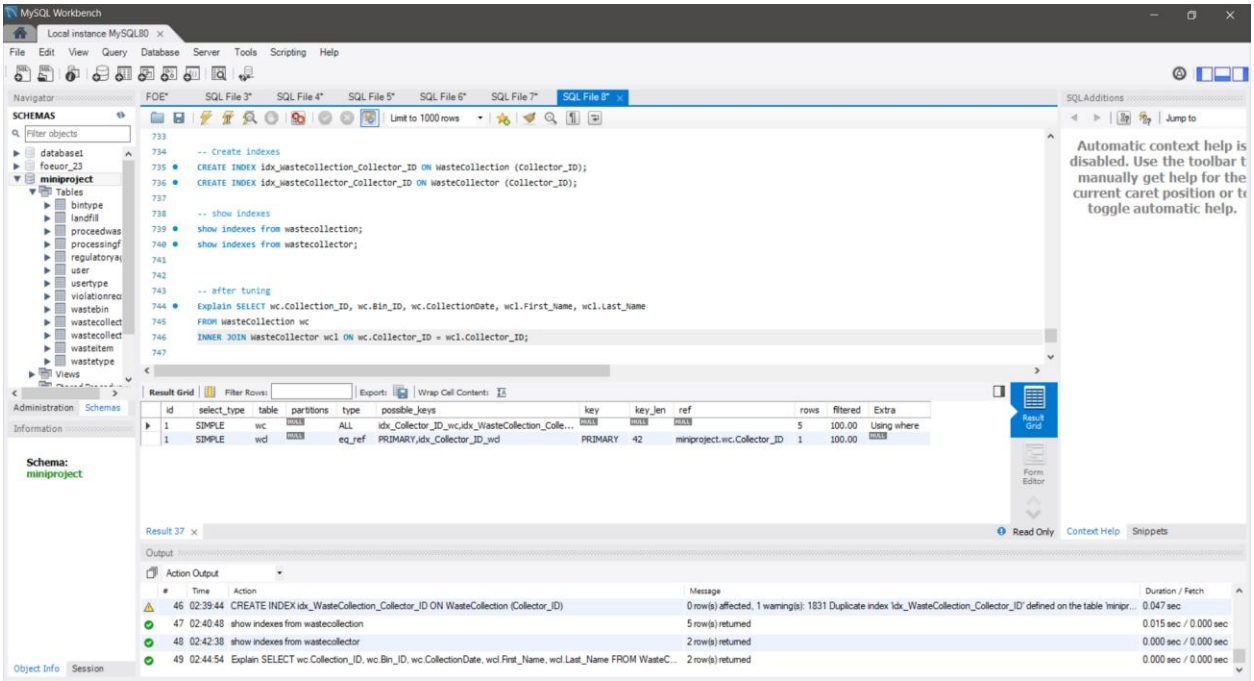
Create index in WasteCollection



Create index in WasteCollector

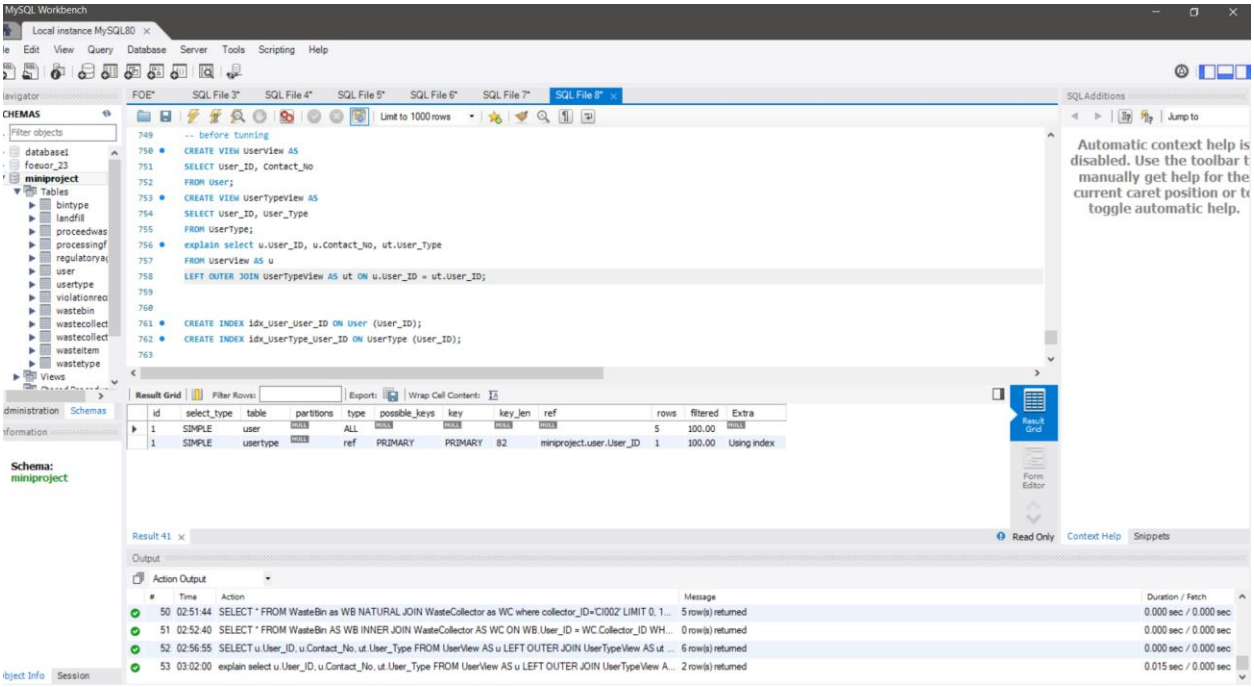


After Tunning

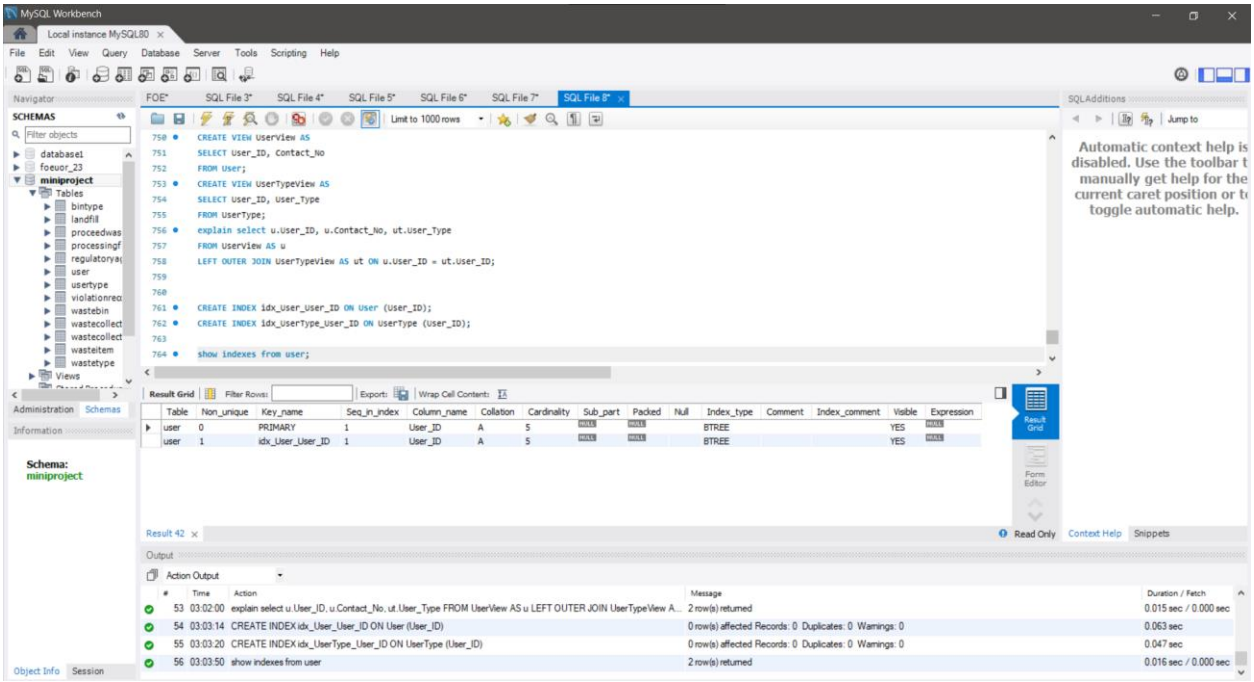


Left Outer Join

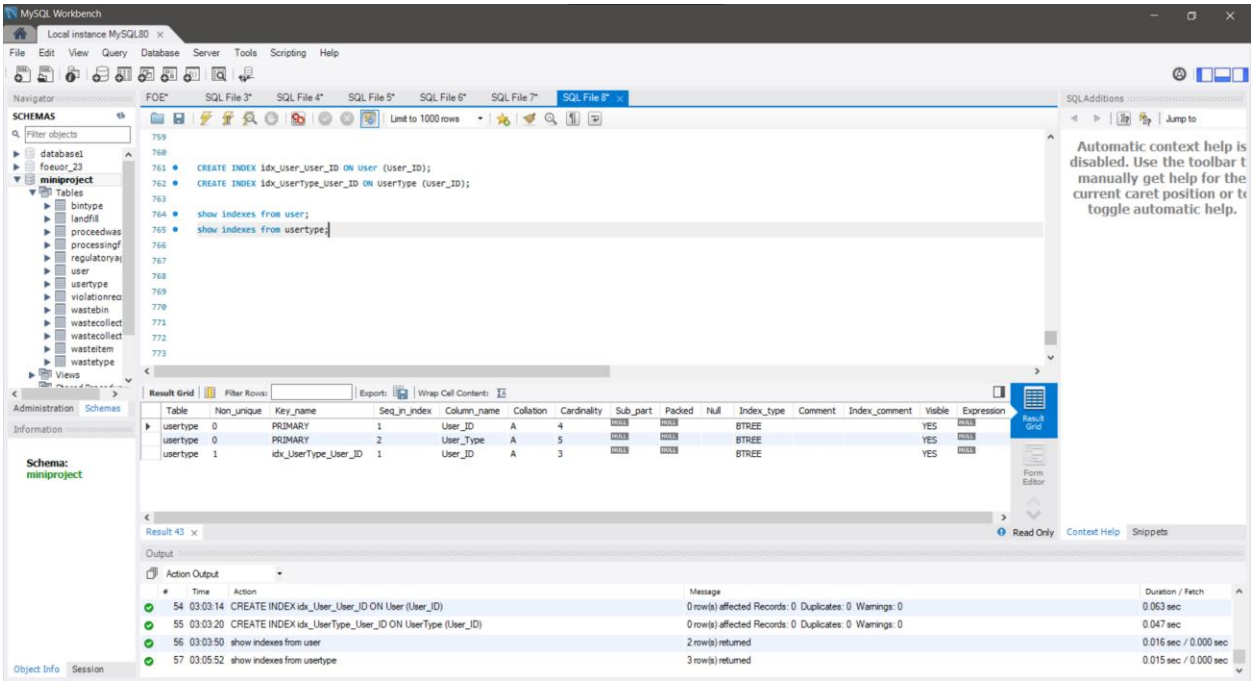
Before Tunning



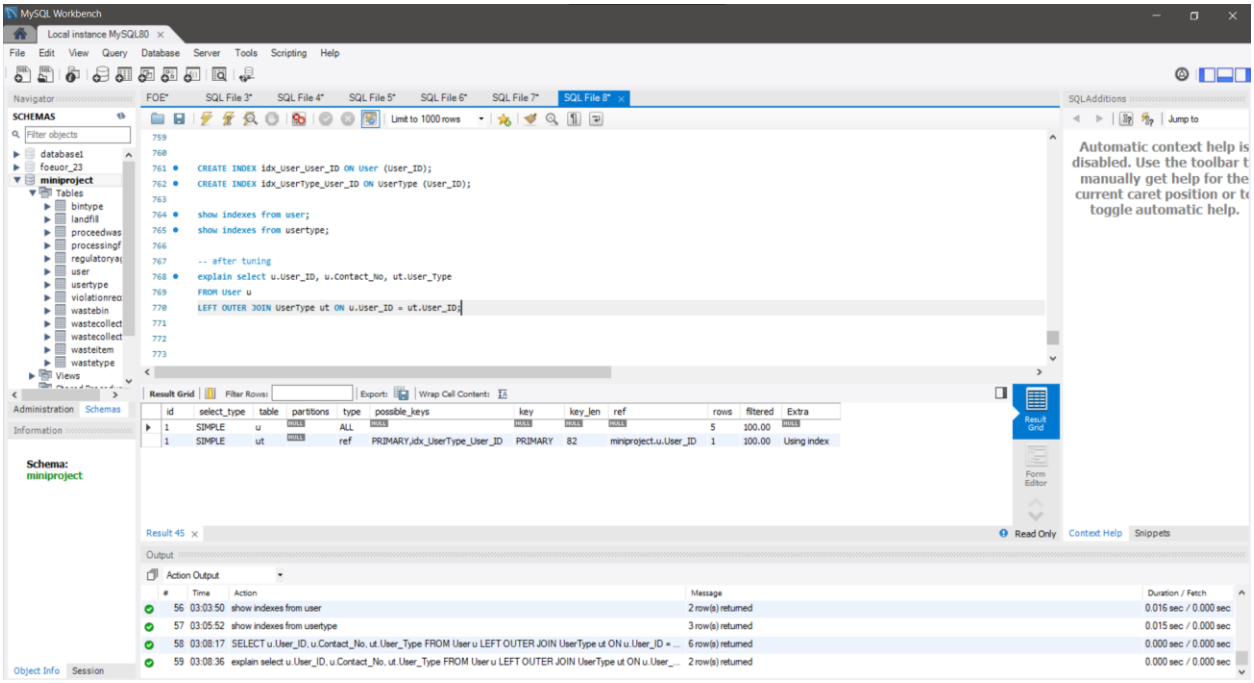
Create index in User



Create index in UserType

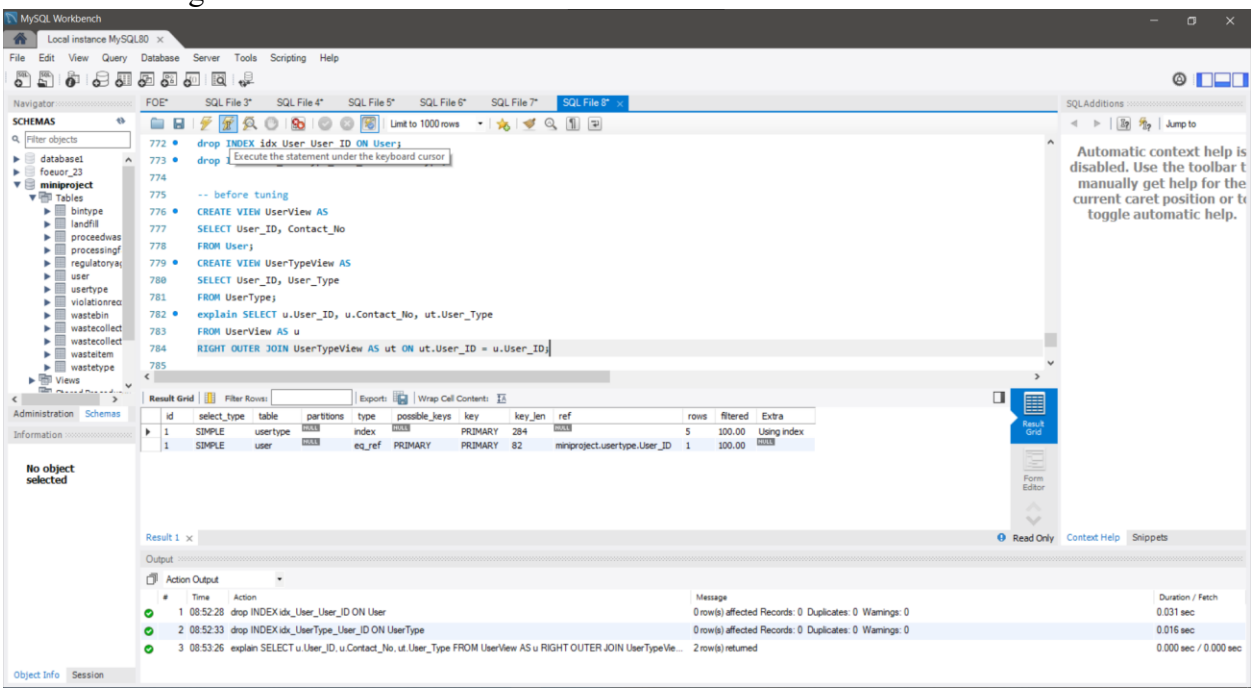


After Tunning

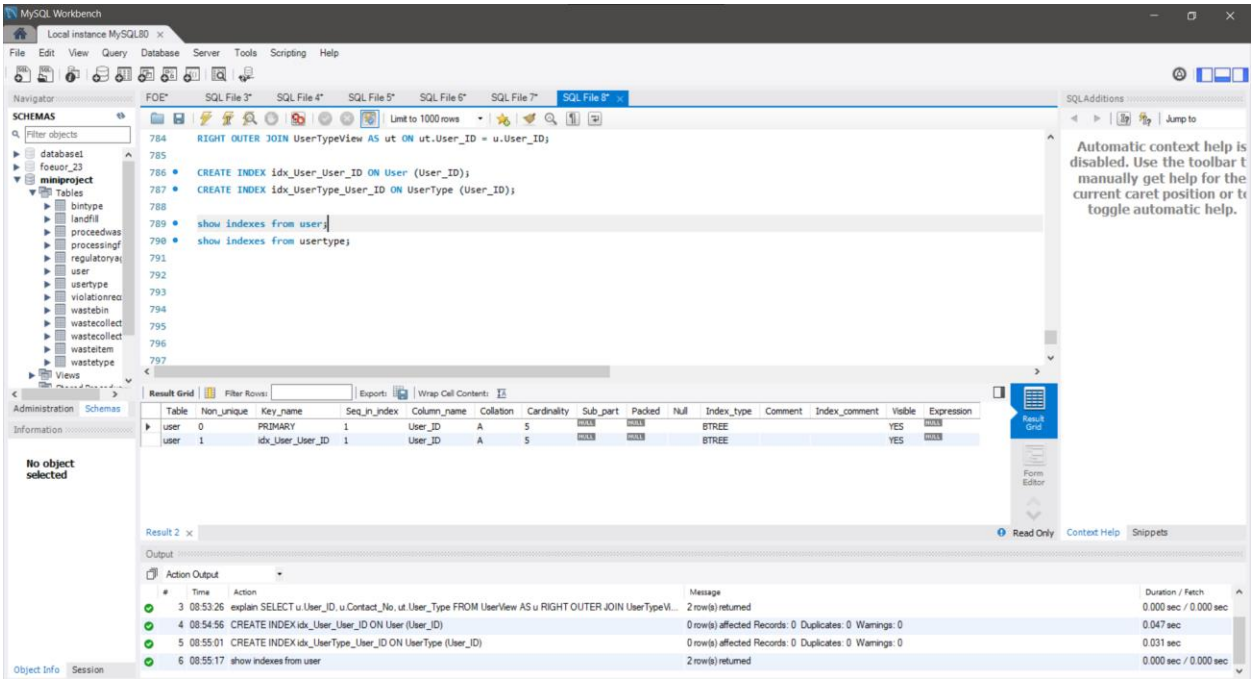


Right Outer Join

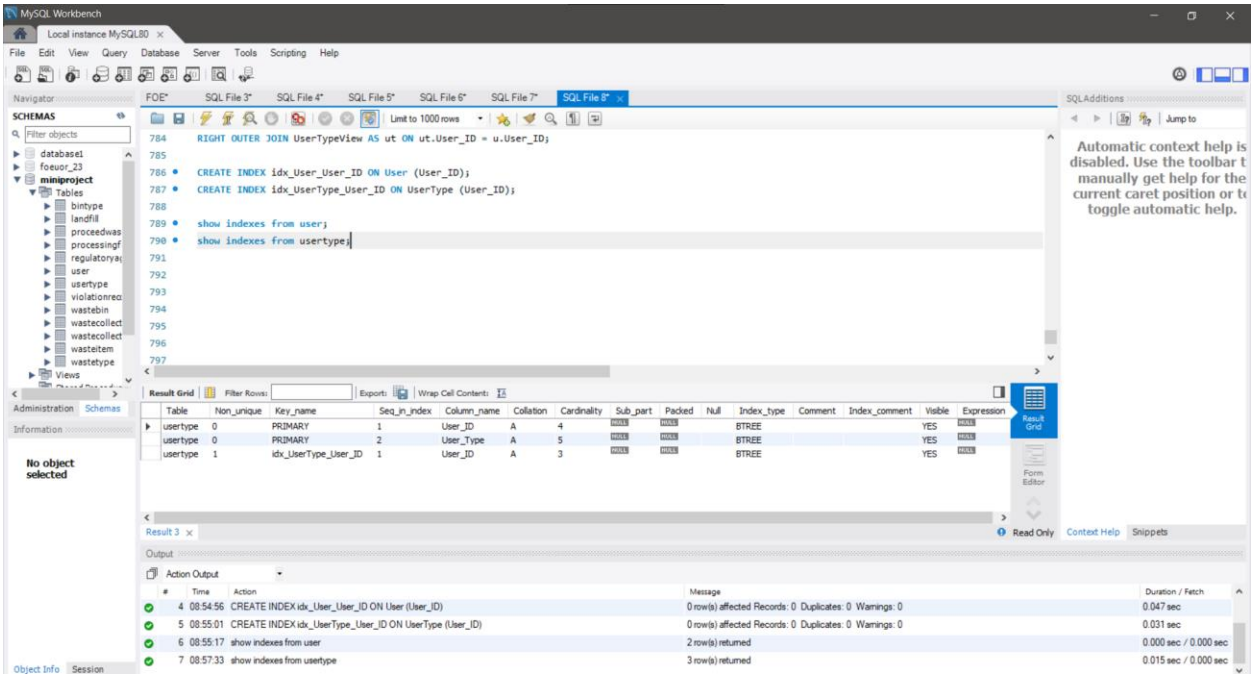
Before Tunning



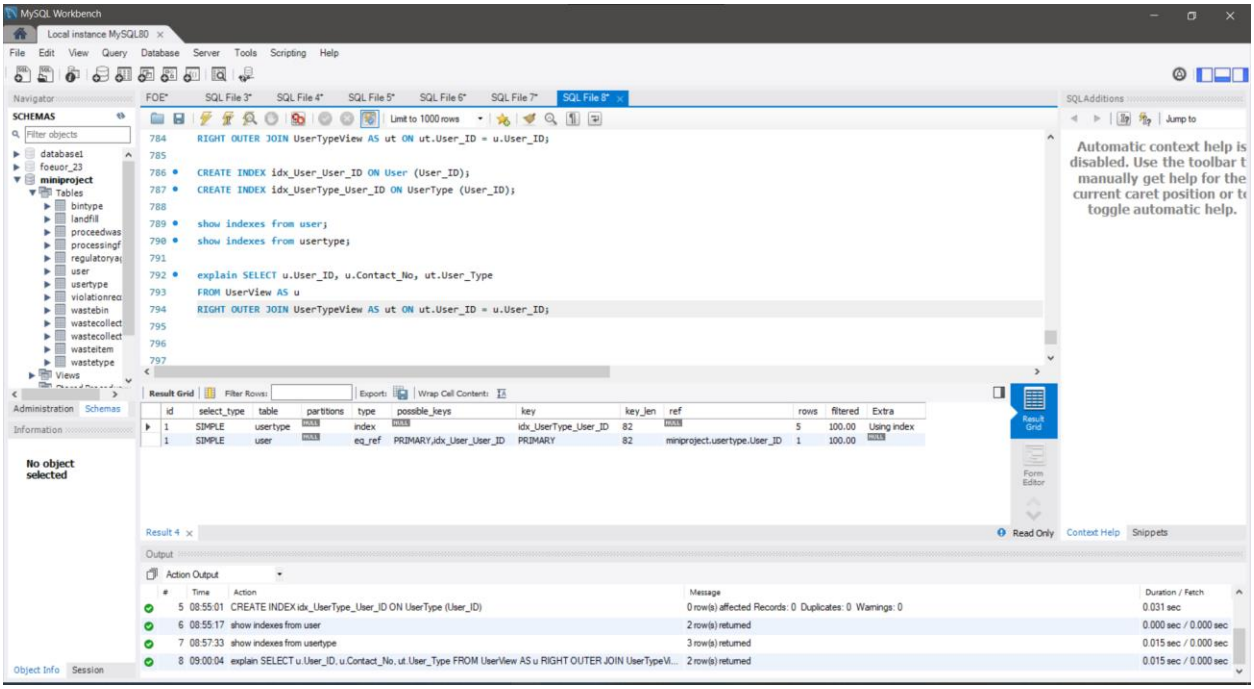
Show index in User



Create index in UserType

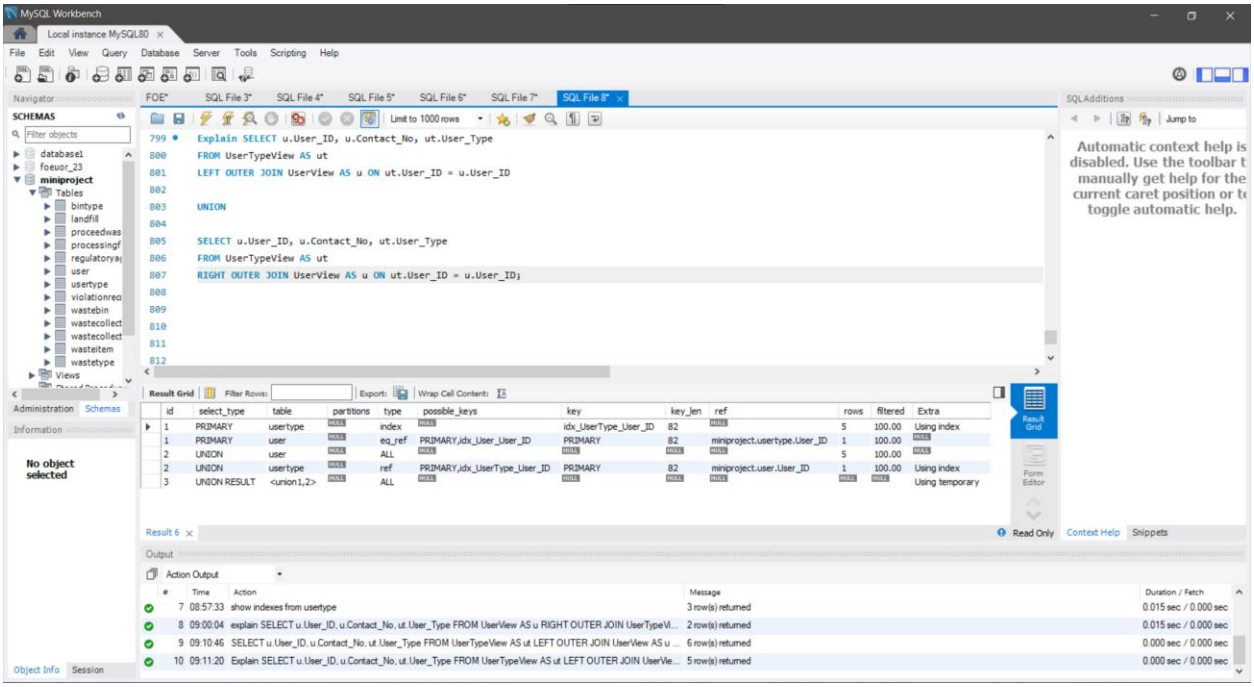


After Tunning

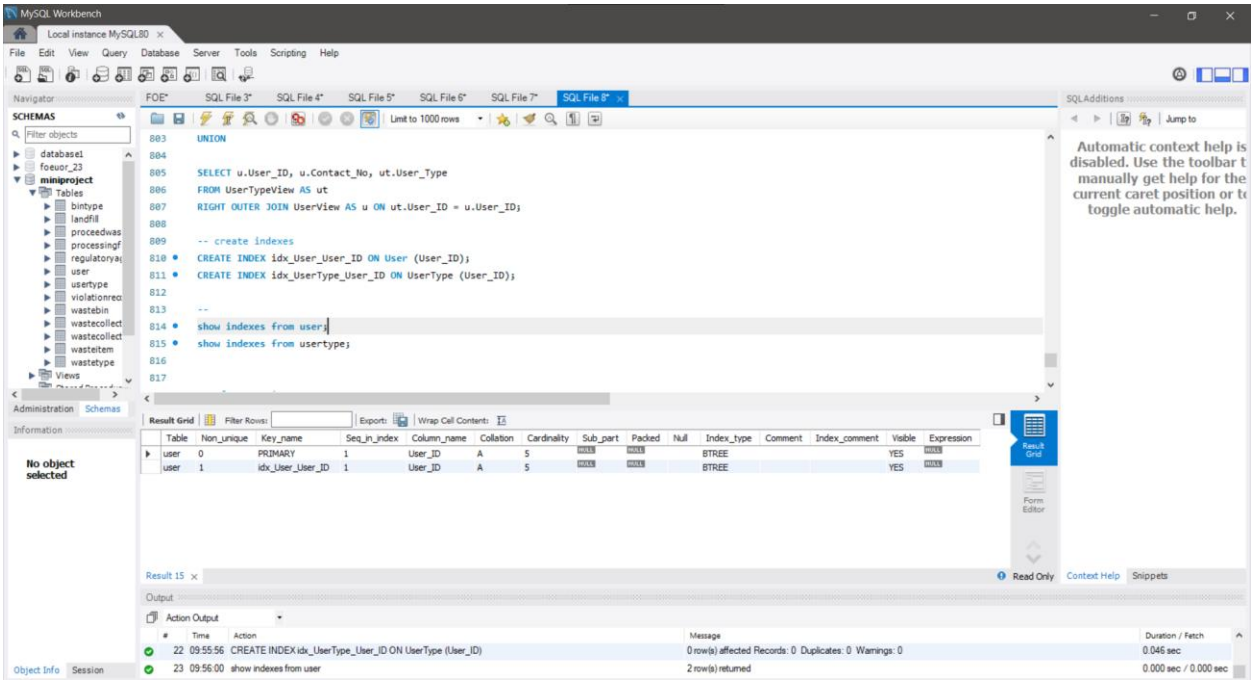


Full outer join

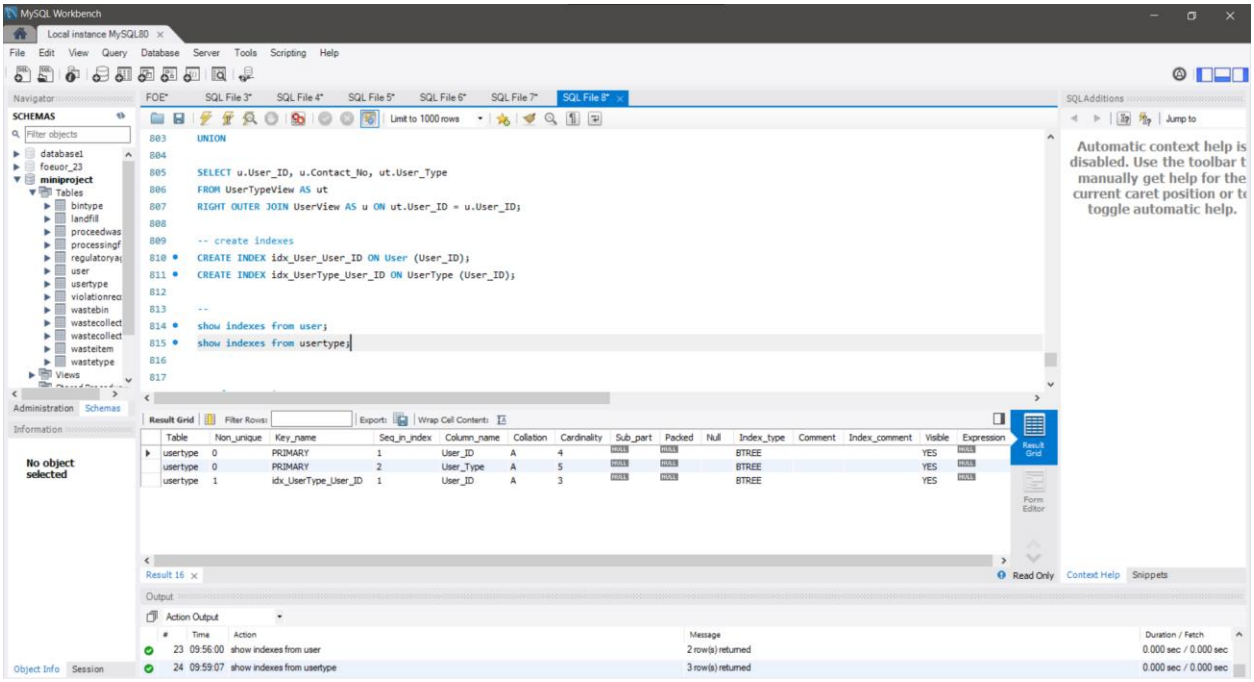
Before Tunning



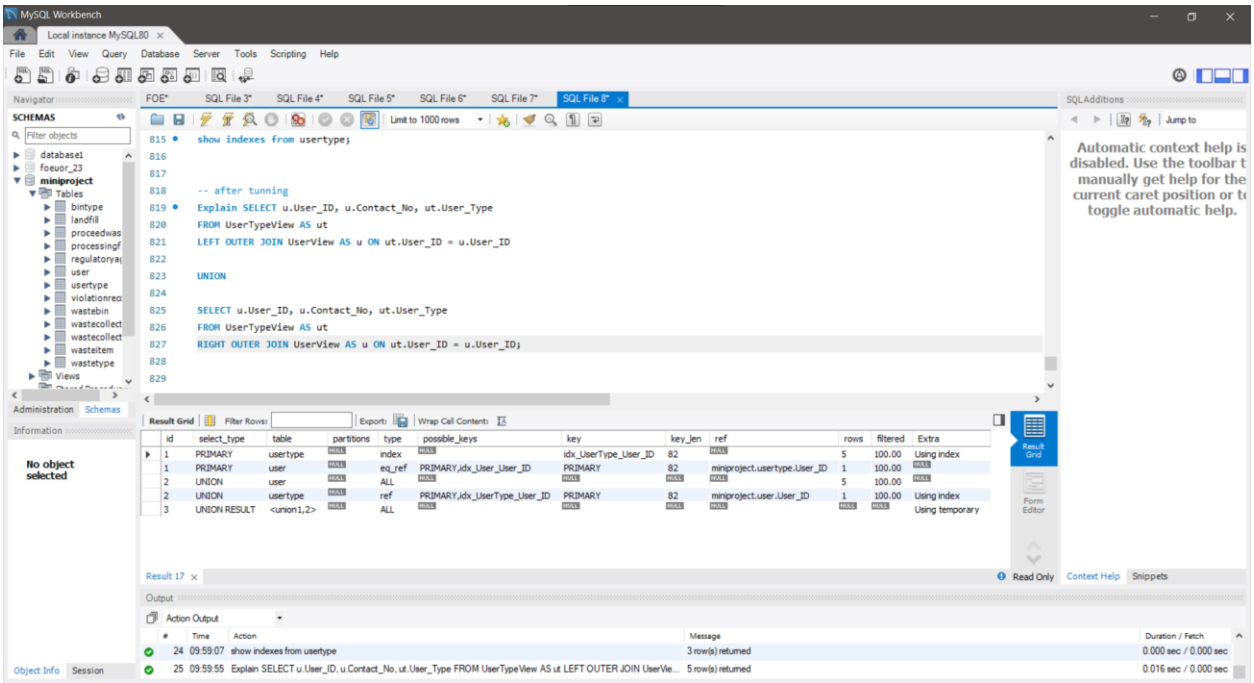
Create index in User



Create index in UserType

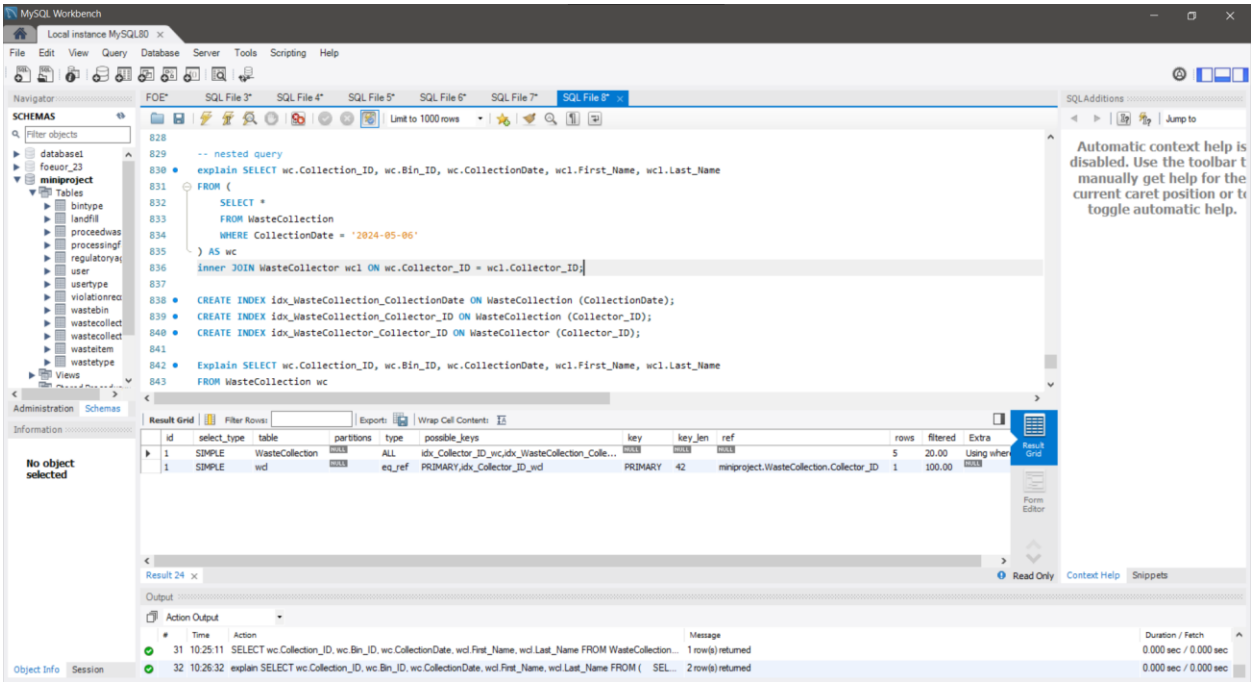


After tunning

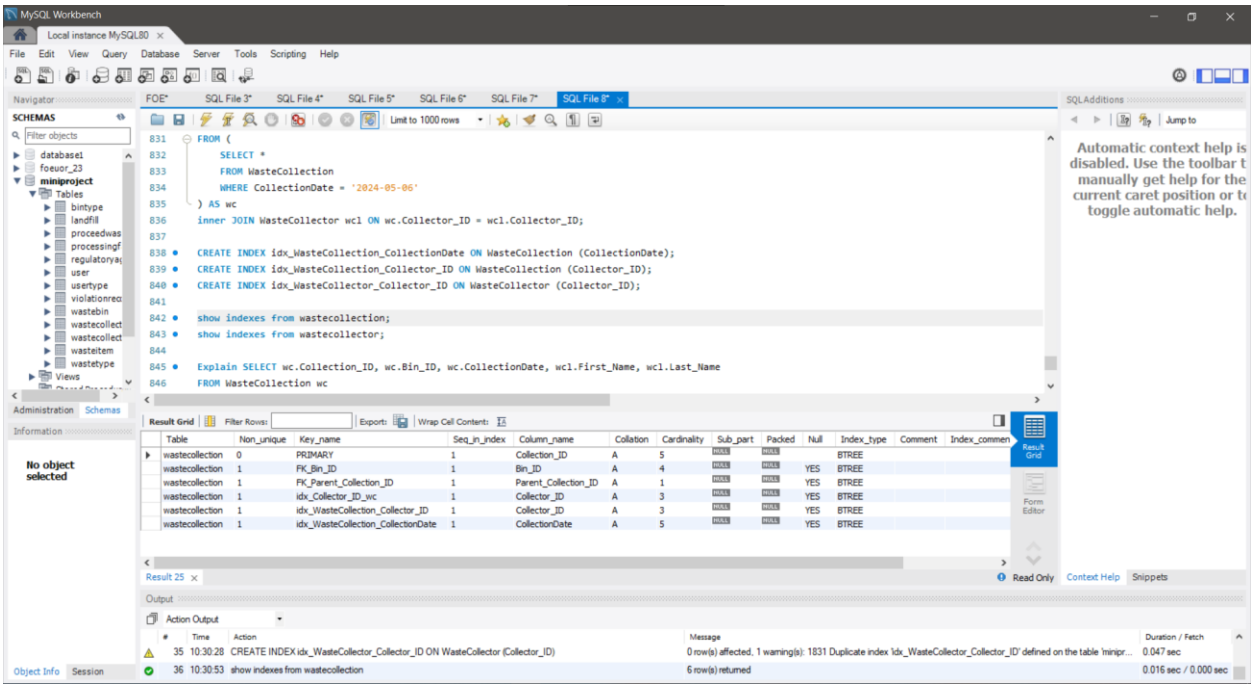


Nested Query with inner Join

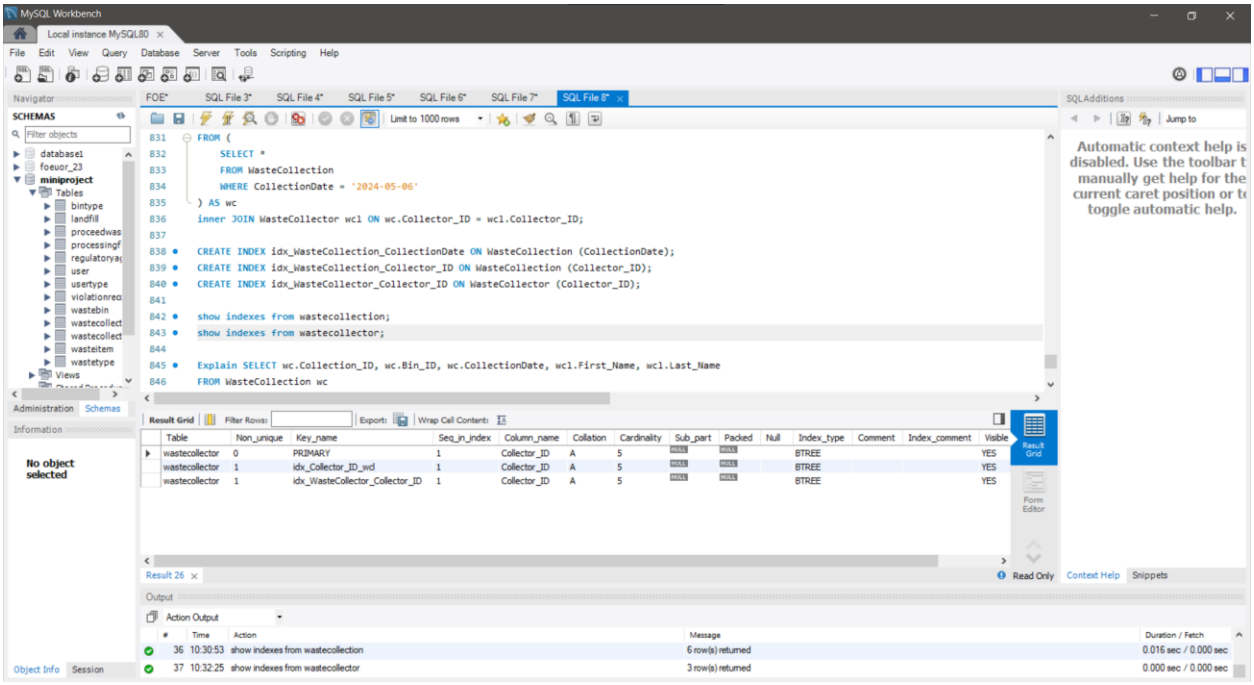
Before Tunning



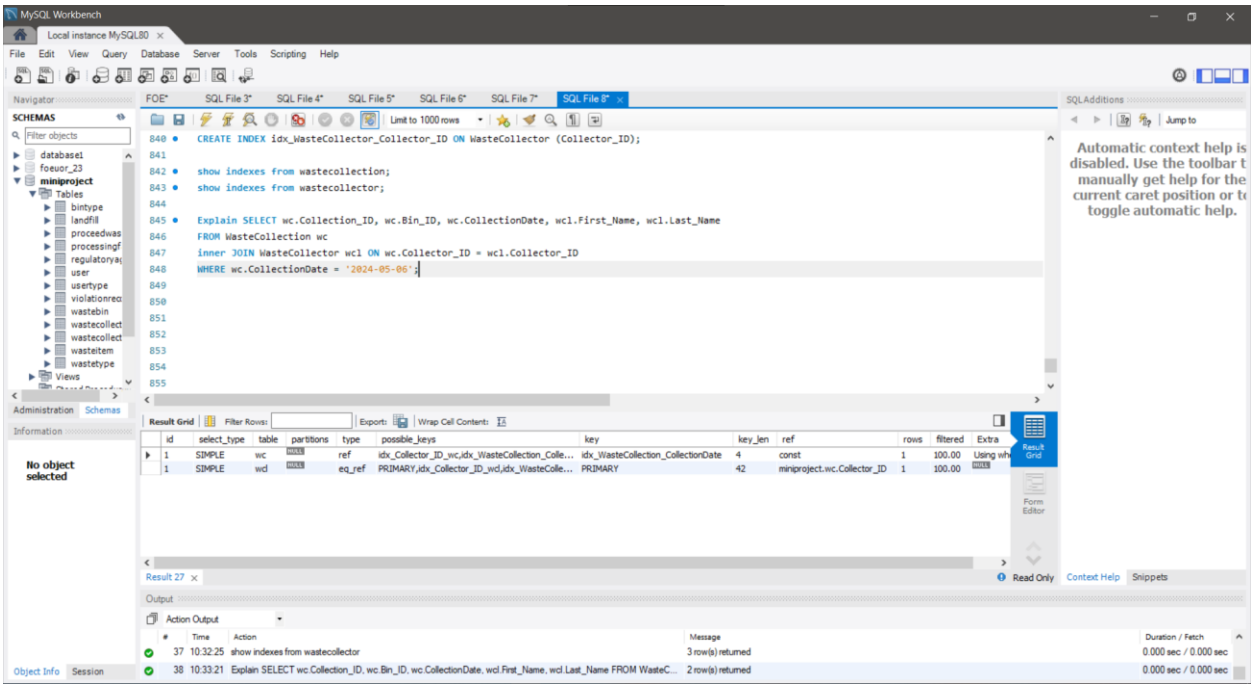
Create index in WasteCollection



Create index in Waste Collection

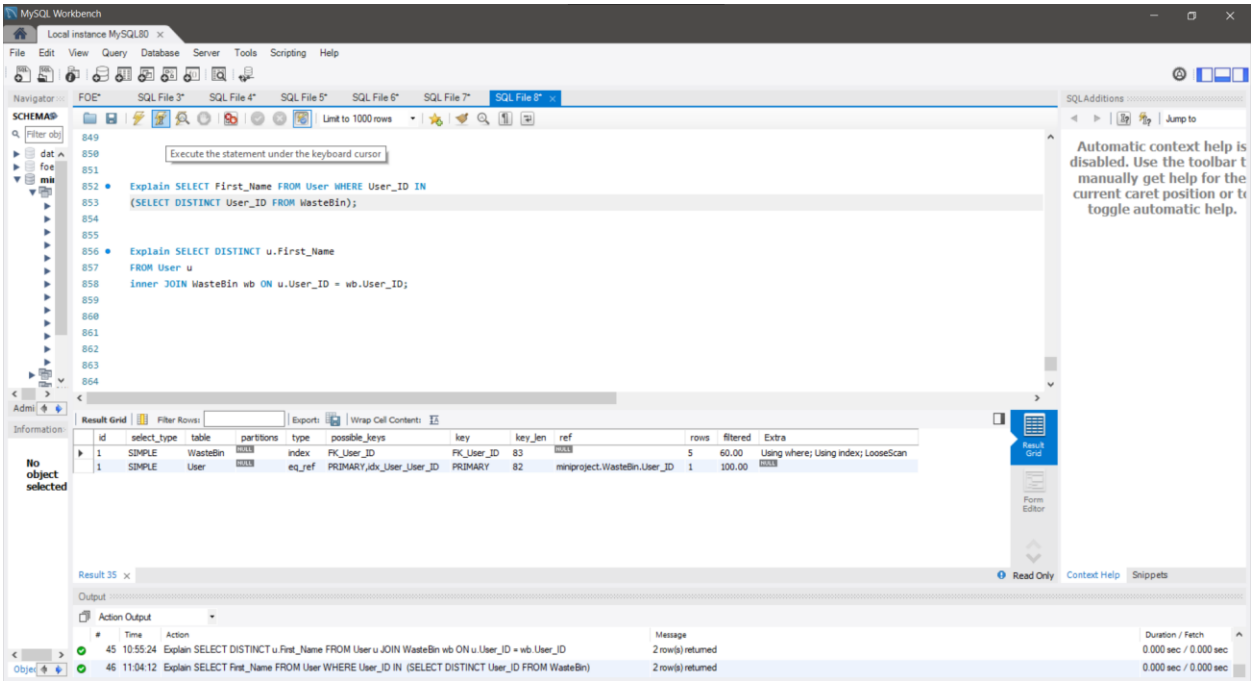


Aftre Tunning

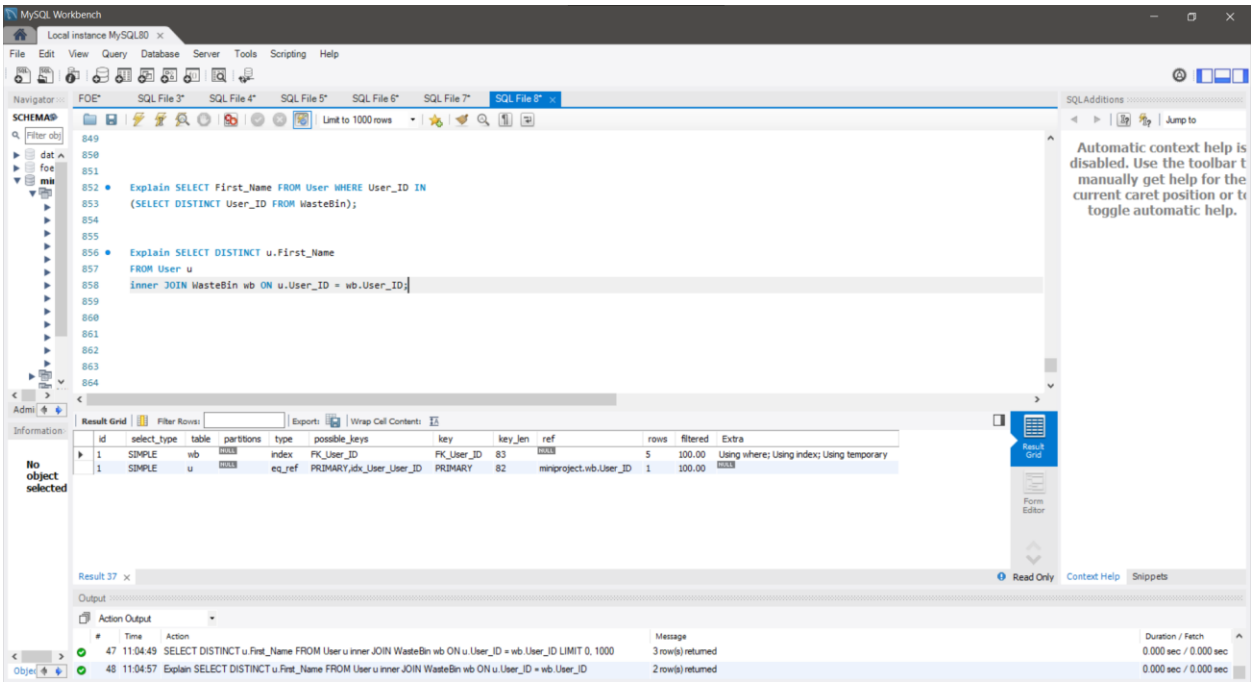


Nested Query with Projection

Before Tunning

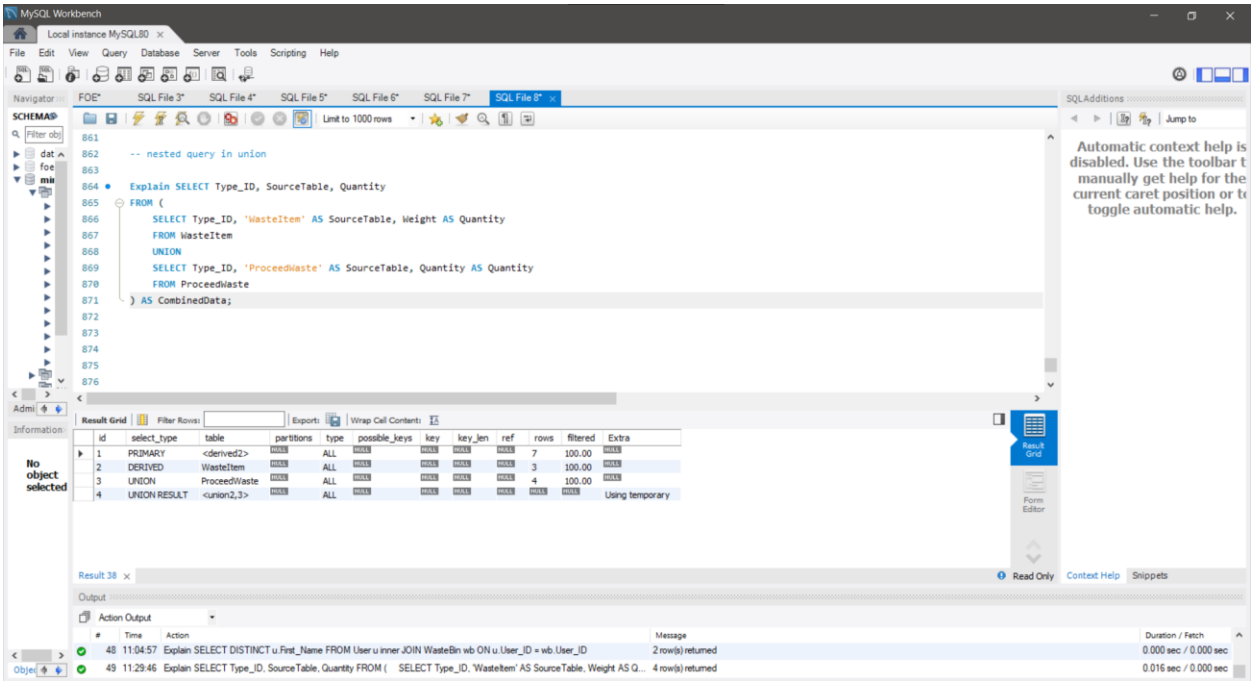


After Tuning

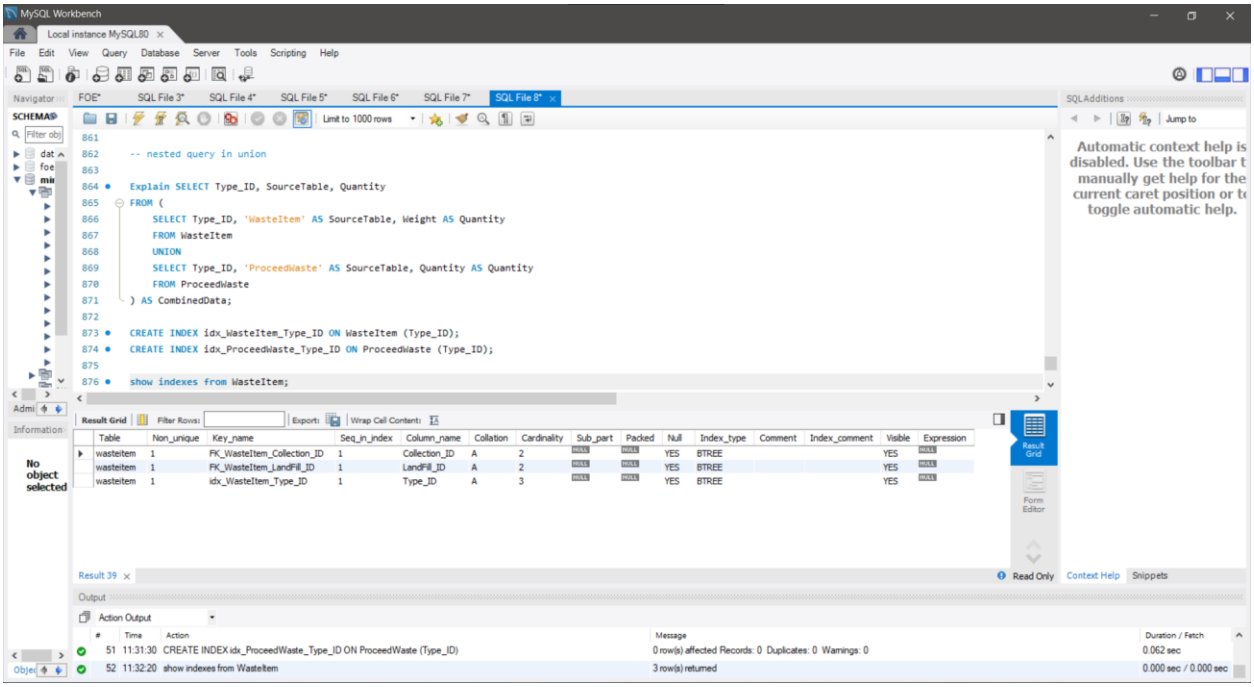


Nested Query with Union

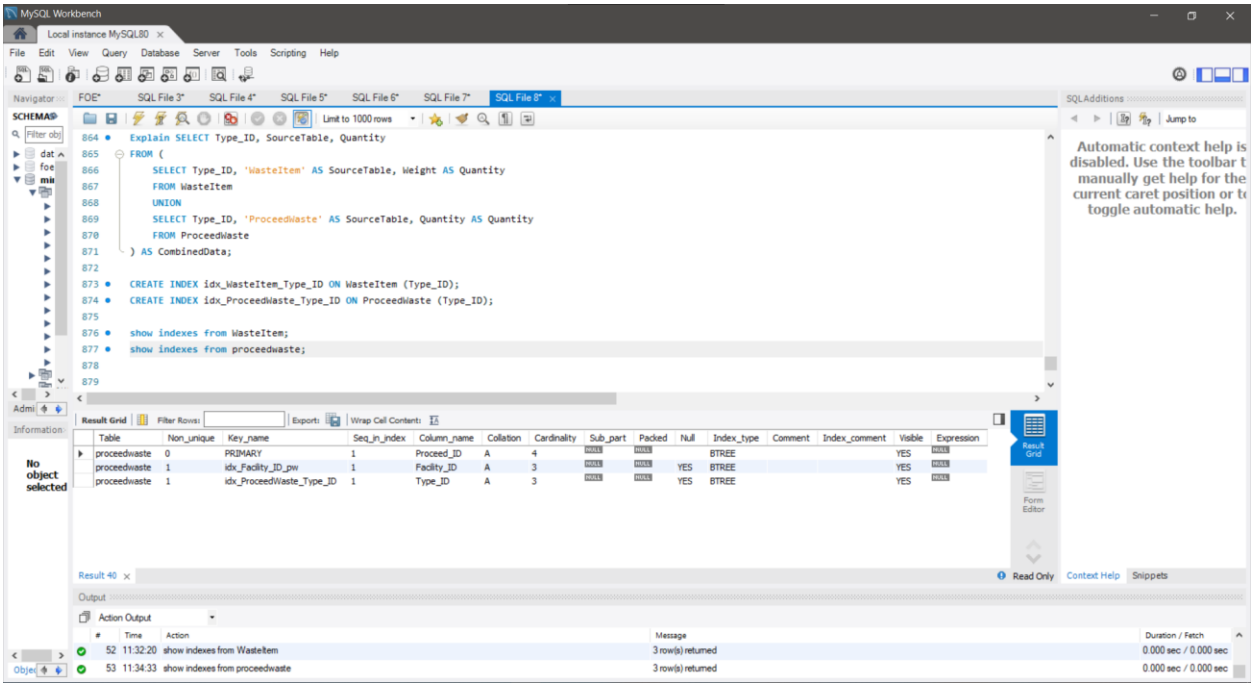
Before Tunning



Create Index WasteItem



Create Index in ProceedWaste



After Tuning