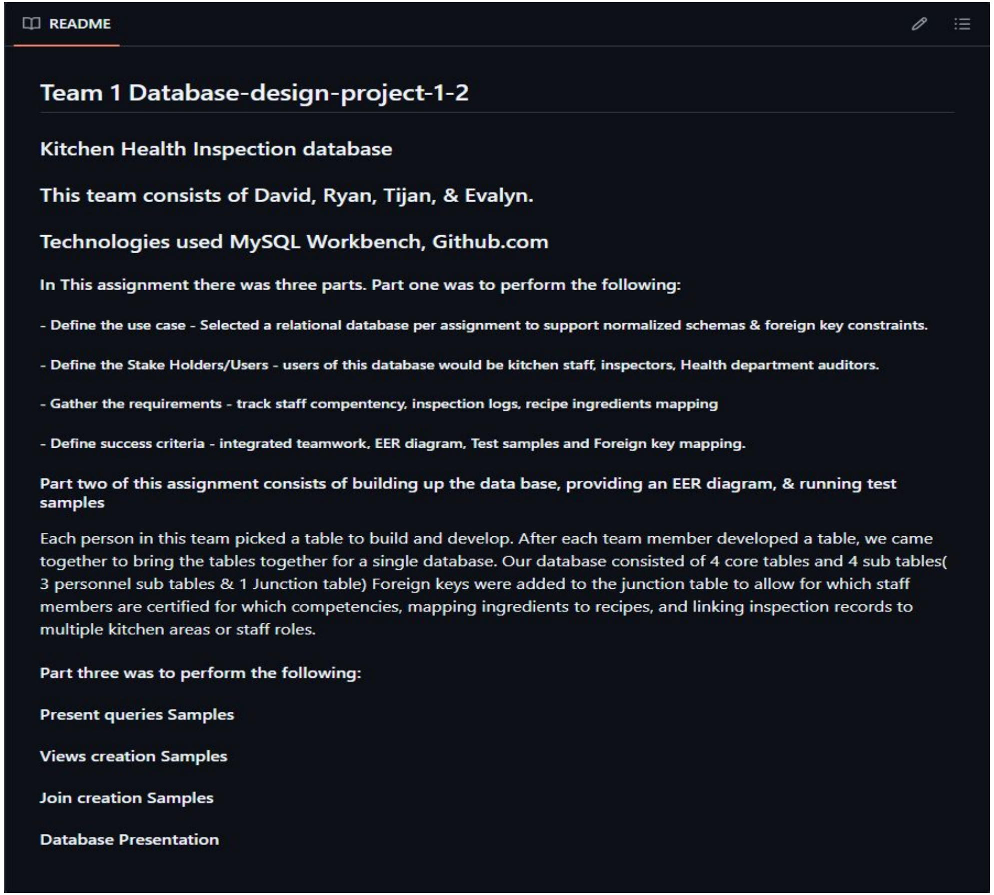
Restaurant Health Inspection

Database For Maintaining a Healthy Workplace

Repository Link

Barrcardavis/IT125-Database-design-project-1-2: Kitchen Personel tables

Our Database



Team 1 Database-design-project-1-2

Kitchen Health Inspection database

This team consists of David, Ryan, Tijan, & Evalyn.

Technologies used MySQL Workbench, Github.com

In This assignment there was three parts. Part one was to perform the following:

- Define the use case - Selected a relational database per assignment to support normalized schemas & foreign key constraints.

- Define the Stake Holders/Users - users of this database would be kitchen staff, inspectors, Health department auditors.

- Gather the requirements - track staff compentency, inspection logs, recipe ingredients mapping

- Define success criteria - integrated teamwork, EER diagram, Test samples and Foreign key mapping.

Part two of this assignment consists of building up the data base, providing an EER diagram, & running test samples

Each person in this team picked a table to build and develop. After each team member developed a table, we came together to bring the tables together for a single database. Our database consisted of 4 core tables and 4 sub tables(3 personnel sub tables & 1 Junction table) Foreign keys were added to the junction table to allow for which staff members are certified for which competencies, mapping ingredients to recipes, and linking inspection records to multiple kitchen areas or staff roles.

Part three was to perform the following:

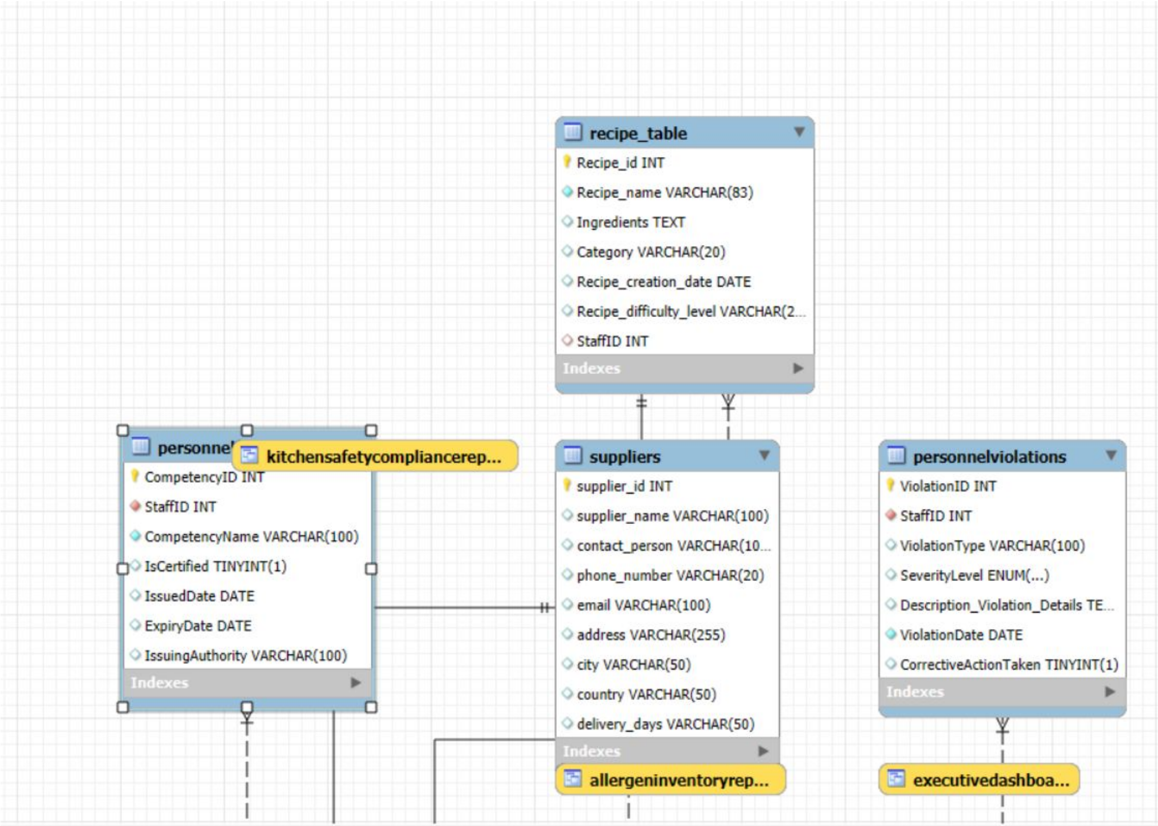
Present queries Samples

Views creation Samples

Join creation Samples

Database Presentation

Kitchen Health Inspection Database Table



recipe_table
- Recipe_id INT
- Recipe_name VARCHAR(83)
- Ingredients TEXT
- Category VARCHAR(20)
- Recipe_creation_date DATE
- Recipe_difficulty_level VARCHAR(2...
- StaffID INT

Indexes

personne
- CompetencyID INT
- StaffID INT
- CompetencyName VARCHAR(100)
- IsCertified TINYINT(1)
- IssuedDate DATE
- ExpiryDate DATE
- IssuingAuthority VARCHAR(100)

Indexes

kitchensafetycompliancerep...

suppliers
- supplier_id INT
- supplier_name VARCHAR(100)
- contact_person VARCHAR(10...
- phone_number VARCHAR(20)
- email VARCHAR(100)
- address VARCHAR(255)
- city VARCHAR(50)
- country VARCHAR(50)
- delivery_days VARCHAR(50)

Indexes

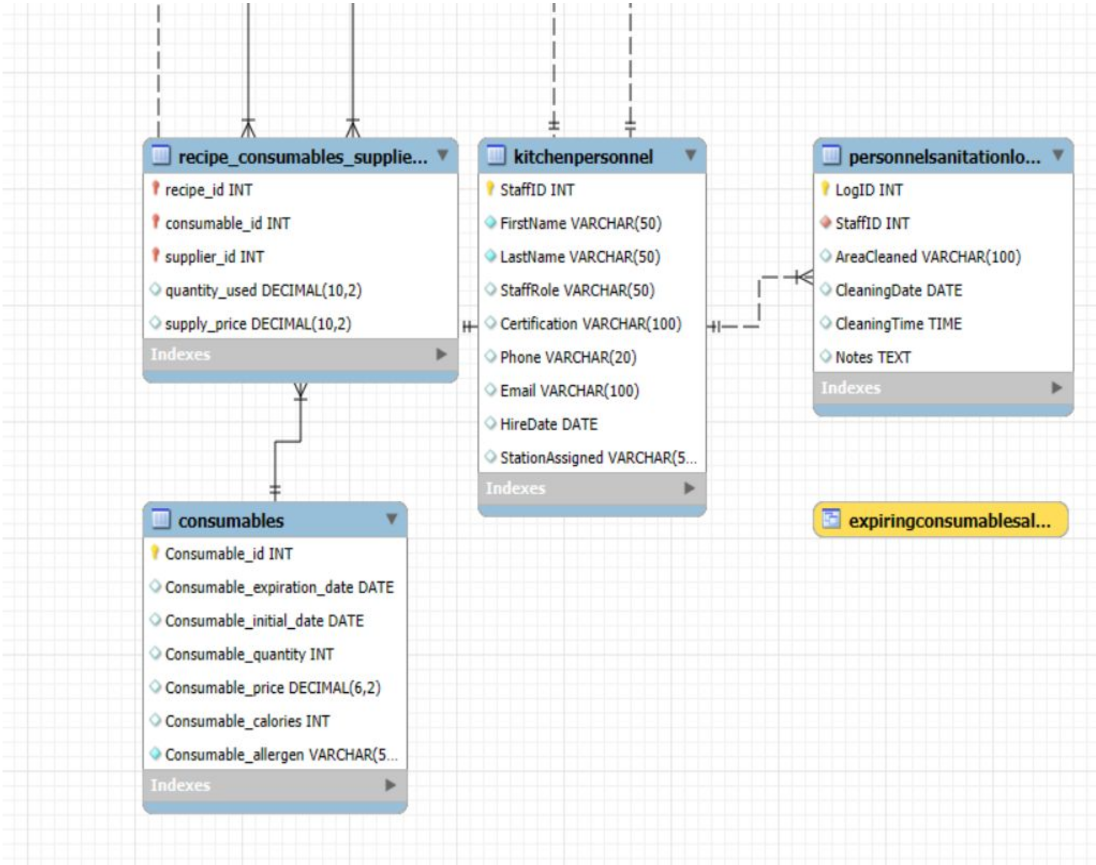
allergeninventoryrep...

personnelviolations
- ViolationID INT
- StaffID INT
- ViolationType VARCHAR(100)
- SeverityLevel ENUM(...)
- Description_Violation_Details TE...
- ViolationDate DATE
- CorrectiveActionTaken TINYINT(1)

Indexes

executivedashboa...

Kitchen Health Inspection Database Table



recipe_consumables_supplie...
recipe_id INT
consumable_id INT
supplier_id INT
quantity_used DECIMAL(10,2)
supply_price DECIMAL(10,2)
Indexes

kitchenpersonnel
StaffID INT
FirstName VARCHAR(50)
LastName VARCHAR(50)
StaffRole VARCHAR(50)
Certification VARCHAR(100)
Phone VARCHAR(20)
Email VARCHAR(100)
HireDate DATE
StationAssigned VARCHAR(5...
Indexes

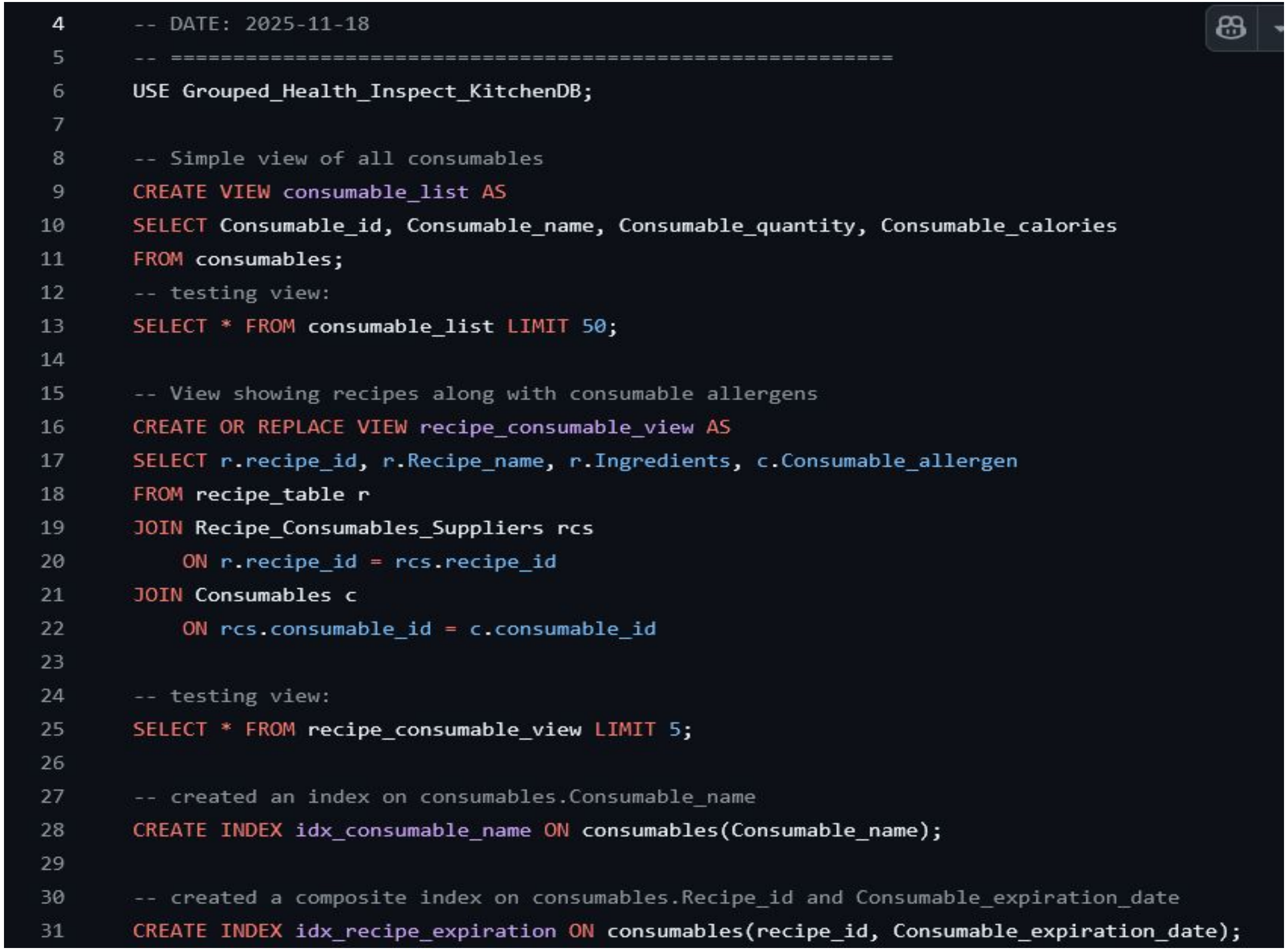
personnelsanitationlo...
LogID INT
StaffID INT
AreaCleaned VARCHAR(100)
CleaningDate DATE
CleaningTime TIME
Notes TEXT
Indexes

consumables
Consumable_id INT
Consumable_expiration_date DATE
Consumable_initial_date DATE
Consumable_quantity INT
Consumable_price DECIMAL(6,2)
Consumable_calories INT
Consumable_allergen VARCHAR(5...
Indexes

expiringconsumablesal...

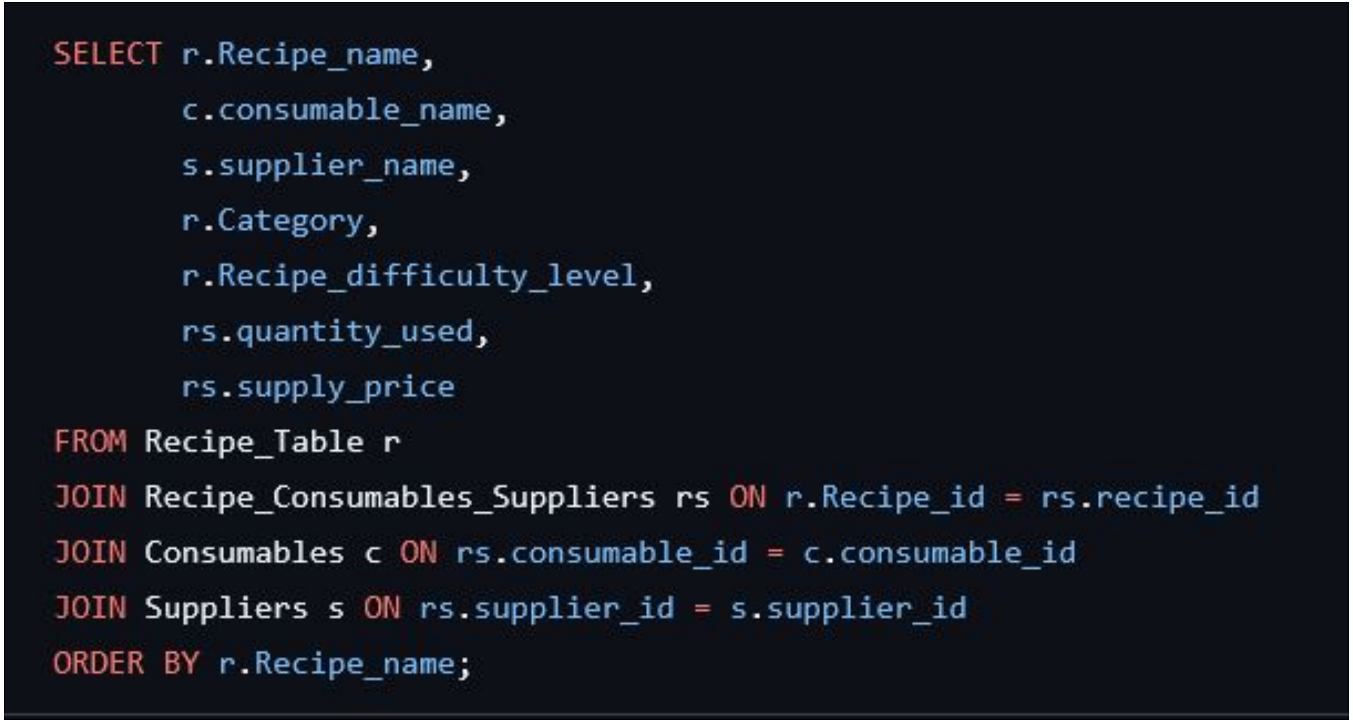
Quarterly Database Project Part 3

Query Sample Datas



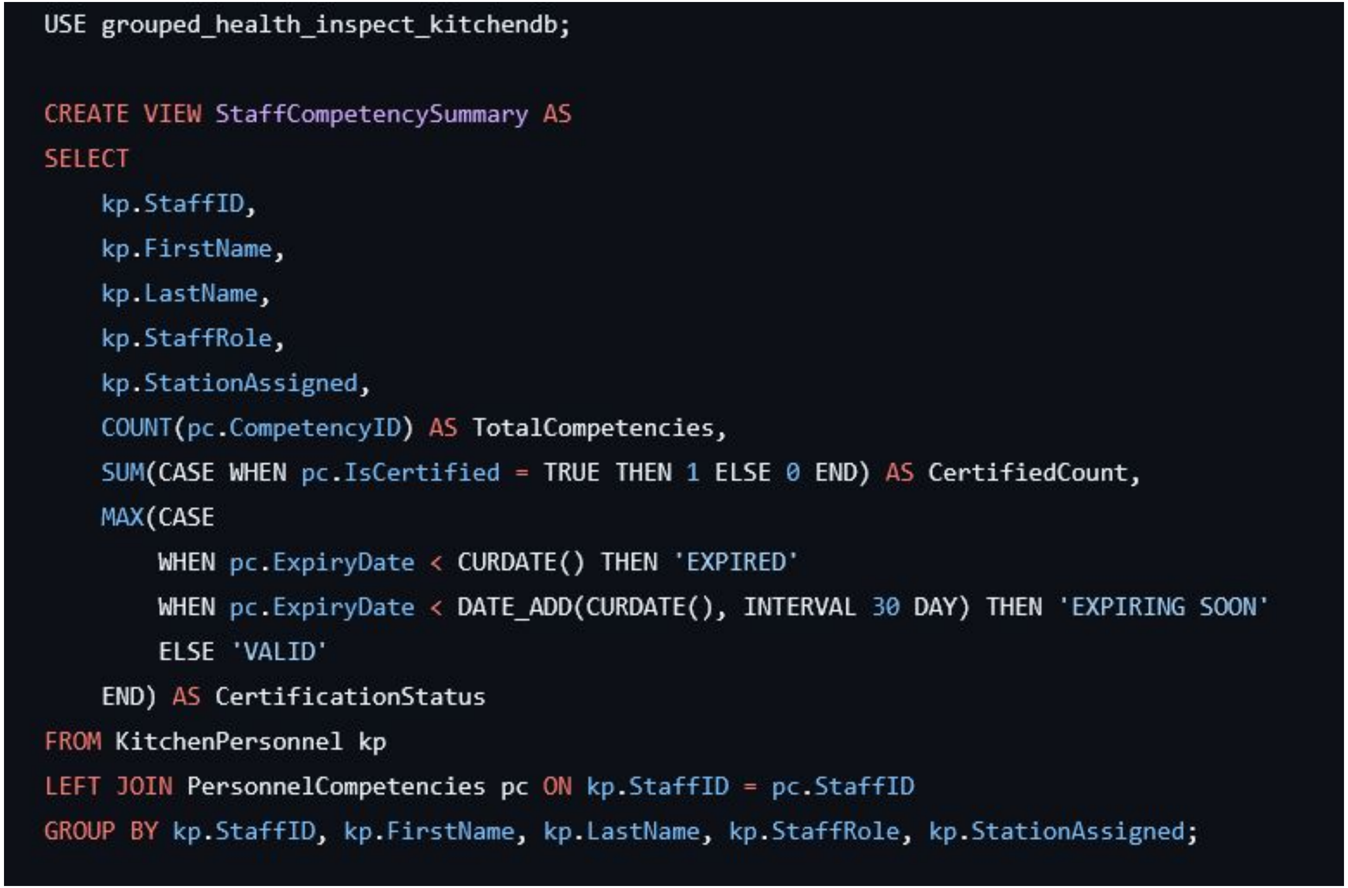
```sql
-- DATE: 2025-11-18
-- ==========================================================
USE Grouped_Health_Inspect_KitchenDB;

-- Simple view of all consumables
CREATE VIEW consumable_list AS
SELECT Consumable_id, Consumable_name, Consumable_quantity, Consumable_calories
FROM consumables;
-- testing view:
SELECT * FROM consumable_list LIMIT 50;

-- View showing recipes along with consumable allergens
CREATE OR REPLACE VIEW recipe_consumable_view AS
SELECT r.recipe_id, r.Recipe_name, r.Ingredients, c.Consumable_allergen
FROM recipe_table r
JOIN Recipe_Consumables_Suppliers rcs
    ON r.recipe_id = rcs.recipe_id
JOIN Consumables c
    ON rcs.consumable_id = c.consumable_id

-- testing view:
SELECT * FROM recipe_consumable_view LIMIT 5;

-- created an index on consumables.Consumable_name
CREATE INDEX idx_consumable_name ON consumables(Consumable_name);

-- created a composite index on consumables.Recipe_id and Consumable_expiration_date
CREATE INDEX idx_recipe_expiration ON consumables(recipe_id, Consumable_expiration_date);
```

Join Sample



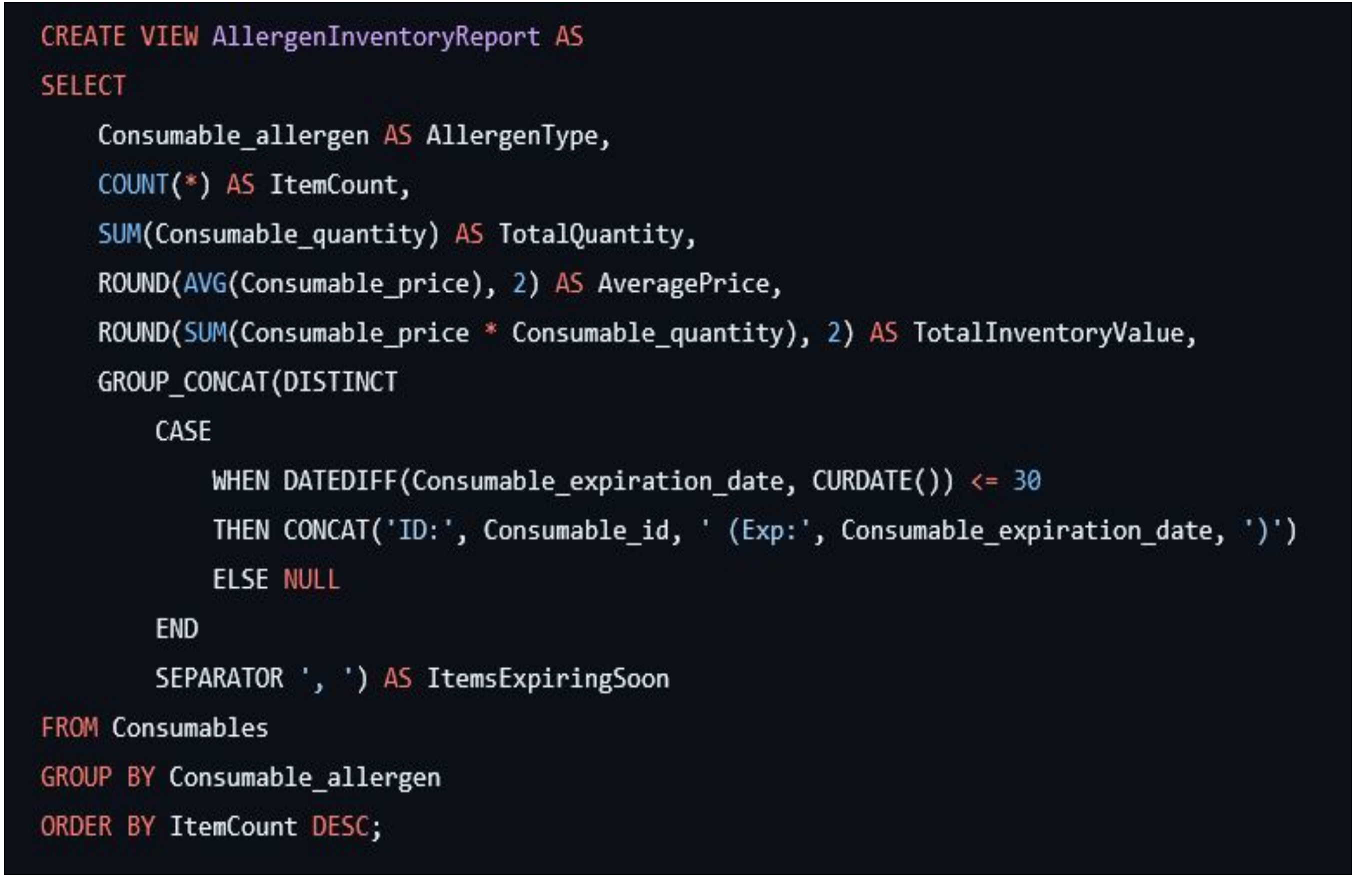
```sql
SELECT r.Recipe_name,
       c.consumable_name,
       s.supplier_name,
       r.Category,
       r.Recipe_difficulty_level,
       rs.quantity_used,
       rs.supply_price
FROM Recipe_Table r
JOIN Recipe_Consumables_Suppliers rs ON r.Recipe_id = rs.recipe_id
JOIN Consumables c ON rs.consumable_id = c.consumable_id
JOIN Suppliers s ON rs.supplier_id = s.supplier_id
ORDER BY r.Recipe_name;
```

Create view Sample



```
USE grouped_health_inspect_kitchendb;

CREATE VIEW StaffCompetencySummary AS
SELECT
    kp.StaffID,
    kp.FirstName,
    kp.LastName,
    kp.StaffRole,
    kp.StationAssigned,
    COUNT(pc.CompetencyID) AS TotalCompetencies,
    SUM(CASE WHEN pc.IsCertified = TRUE THEN 1 ELSE 0 END) AS CertifiedCount,
    MAX(CASE
        WHEN pc.ExpiryDate < CURDATE() THEN 'EXPIRED'
        WHEN pc.ExpiryDate < DATE_ADD(CURDATE(), INTERVAL 30 DAY) THEN 'EXPIRING SOON'
        ELSE 'VALID'
    END) AS CertificationStatus
FROM KitchenPersonnel kp
LEFT JOIN PersonnelCompetencies pc ON kp.StaffID = pc.StaffID
GROUP BY kp.StaffID, kp.FirstName, kp.LastName, kp.StaffRole, kp.StationAssigned;
```

Create View Sample



```sql
CREATE VIEW AllergenInventoryReport AS
SELECT
    Consumable_allergen AS AllergenType,
    COUNT(*) AS ItemCount,
    SUM(Consumable_quantity) AS TotalQuantity,
    ROUND(AVG(Consumable_price), 2) AS AveragePrice,
    ROUND(SUM(Consumable_price * Consumable_quantity), 2) AS TotalInventoryValue,
    GROUP_CONCAT(DISTINCT
        CASE
            WHEN DATEDIFF(Consumable_expiration_date, CURDATE()) <= 30
            THEN CONCAT('ID:', Consumable_id, ' (Exp:', Consumable_expiration_date, ')')
            ELSE NULL
        END
        SEPARATOR ', ') AS ItemsExpiringSoon
FROM Consumables
GROUP BY Consumable_allergen
ORDER BY ItemCount DESC;
```

Lessons Learned and Challenges Faced

- Challenges of working with a team.
- Integrating our separate ideas and the difference in syntax.
- The difference in language as to other coding languages.
- Being open to feedback in the collaboration process.
- Coordinating time to work as a team.
- Lack of communication.
- Our messenger would auto pause notifications. (Slack)
- Time management between classes.
- Returning to school for some of us after a period of time has introduced changes to education structure.