**NAME: NKENGEZA DERICK**

**MATRICLE: FE21A277**

**SQL LAB REPORT**

**Introduction**

The goal of this project is to create a database for customers’ orders and details, and to perform various queries on the data.

The database will contain the following tables:

* Customers: This table will store information about customers, such as their name, address, and country.
* Orders: This table will store information about orders, such as the customer ID, order date, and shipped date.
* Order Details: This table will store information about the products that were ordered, such as the product ID, quantity, and unit price.
* Products: This table will store information about products such as the quantity, price, category, name and its supplier.
* Supplier: This table will store information about the different suppliers that is their names and address.
* Shippers: This table will keep data for the various Shipping companies.

The project will involve creating the database, adding data to the tables, and writing SQL queries, updating, altering and creating new tables.

**Part 1: create database and tables**

CREATE DATABASE orders;

USE orders;

CREATE TABLE orders.tblShippers (

ShipperID INT NOT NULL,

CompanyName VARCHAR(40) NOT NULL UNIQUE,

PRIMARY KEY (ShipperID) );

CREATE TABLE orders.tblCustomers (

CustomerID VARCHAR(5) NOT NULL,

CompanyName VARCHAR(40) NOT NULL,

ContactName VARCHAR(30) NULL,

ContactTitle VARCHAR(30) NULL,

Address VARCHAR(60) NULL,

City VARCHAR(15) NULL,

Region VARCHAR(15) NULL,

PostalCode VARCHAR(10) NULL,

Country VARCHAR(15) NULL DEFAULT 'Canada',

Phone VARCHAR(24) NULL,

Fax VARCHAR(24) NULL,

PRIMARY KEY (CustomerID)

);

CREATE TABLE orders.tblOrders (

OrderID INT NULL,

CustomerID VARCHAR(5) NOT NULL,

EmployeeID INT NULL,

ShipName VARCHAR(40) NULL,

ShipAddress VARCHAR(60) NULL,

ShipCity VARCHAR(15) NULL,

ShipRegion VARCHAR(15) NULL,

ShipPostalCode VARCHAR(10) NULL,

ShipCountry VARCHAR(15) NULL,

ShipVia INT NULL,

OrderDate DATETIME NULL,

RequiredDate DATETIME NULL,

ShippedDate DATETIME NULL,

Freight FLOAT(12, 4) NULL,

CONSTRAINT checkdate CHECK (ShippedDate > OrderDate),

PRIMARY KEY (OrderID),

FOREIGN KEY (CustomerID) REFERENCES tblCustomers (CustomerID) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE orders.tblSupplier (

SupplierID INT NOT NULL,

Name VARCHAR(50) NULL,

Address VARCHAR(50) NULL,

City VARCHAR(50) NULL,

Province VARCHAR(50) NULL,

PRIMARY KEY (SupplierID)

);

CREATE TABLE orders.tblProducts (

ProductID INT NOT NULL,

SupplierID INT NULL,

CategoryID INT NULL,

ProductName VARCHAR(40) NOT NULL,

EnglishName VARCHAR(40) NULL,

QuantityPerUnit VARCHAR(20) NULL,

UnitPrice FLOAT(12, 4) NULL,

UnitsInStock SMALLINT NULL,

UnitsOnOrder SMALLINT NULL,

ReorderLevel SMALLINT NULL,

Discontinued BIT NOT NULL,

PRIMARY KEY (ProductID),

FOREIGN KEY (SupplierID) REFERENCES tblSupplier (SupplierID) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE orders.tblOrderDetails (

OrderID INT NULL,

ProductID INT NULL,

UnitPrice FLOAT(12, 4) NOT NULL,

Quantity SMALLINT NOT NULL ,

Discount REAL NOT NULL,

PRIMARY KEY (OrderID, ProductID),

FOREIGN KEY (OrderID) REFERENCES tblOrders (OrderID) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (ProductID) REFERENCES tblProducts (ProductID) ON DELETE CASCADE ON UPDATE CASCADE

);

**Part 2: SQL Statements**

**Question 1**

SELECT

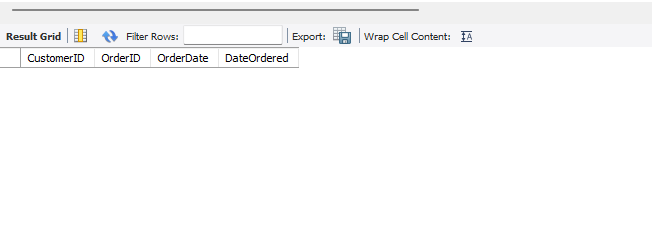
c.CustomerID,

o.OrderID, o.OrderDate, o.RequiredDate as DateOrdered

FROM tblCustomers c

JOIN tblOrders o ON c.CustomerID = o.CustomerID

WHERE o.OrderDate BETWEEN '2014-01-01 00:00:00' AND '2014-12-31 00:00:00';

No row returned

**Question2**

**ALTER TABLE orders.tblCustomers**

**ADD Active BOOLEAN DEFAULT TRUE;**

**Queation 3**

SELECT

c.CompanyName,

o.OrderDate as Date\_Ordered,

SUM( od.UnitPrice\*od.Quantity\*(1-Discount)) + o.Freight AS Total\_Amount

FROM tblOrders o

JOIN tblOrderDetails od

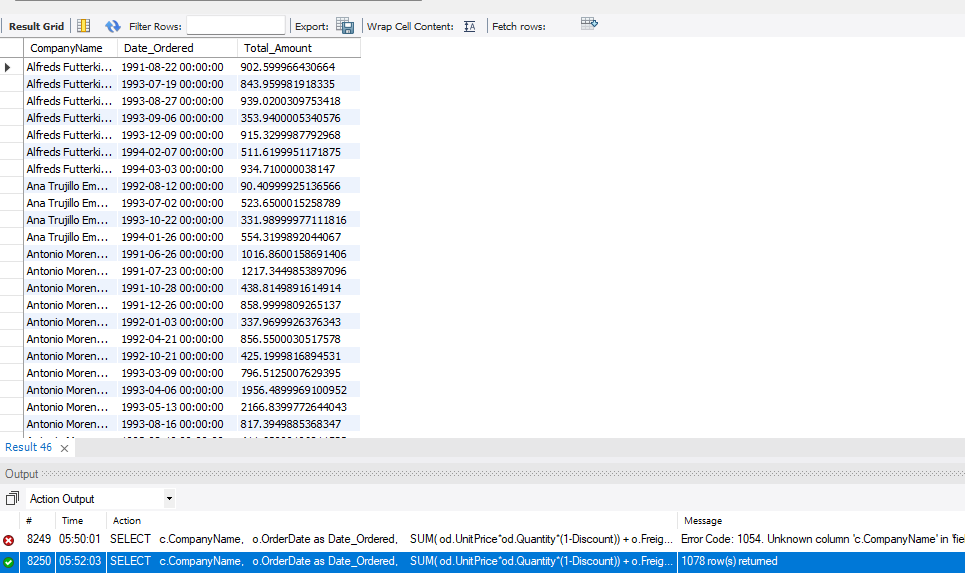
ON od.OrderID = o.OrderID

JOIN tblCustomers c

ON c.CustomerID = o.CustomerID

WHERE o.OrderDate < '2012-09-01 00:00:00'

GROUP BY c.CustomerID, o.OrderDate, o.Freight;

1078 rows returned

**Question 4**

SELECT

o.OrderID, o.ShipName, o.ShipAddress,

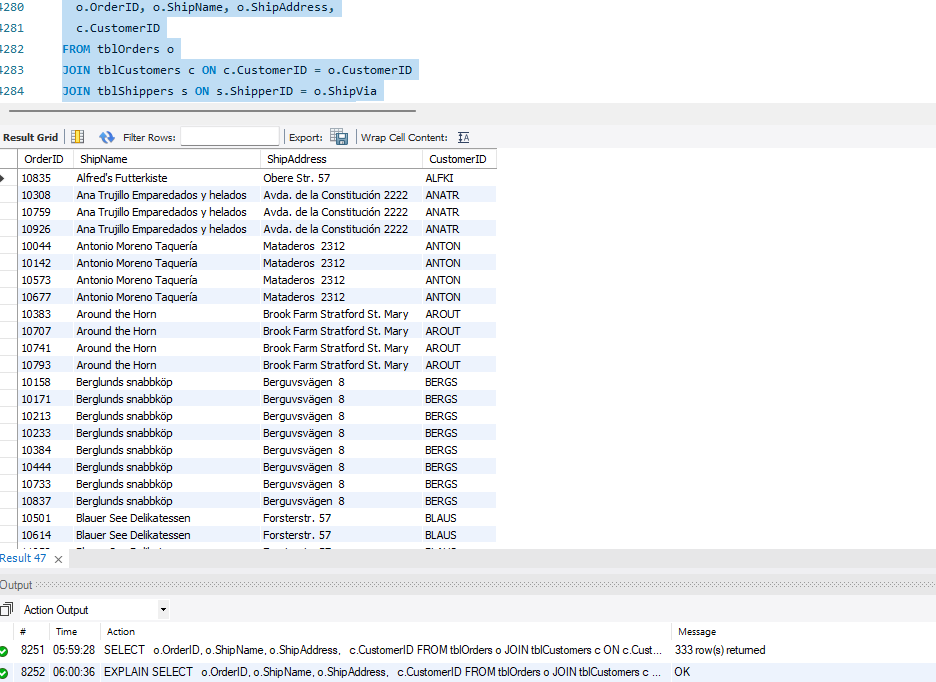
c.CustomerID

FROM tblOrders o

JOIN tblCustomers c ON c.CustomerID = o.CustomerID

JOIN tblShippers s ON s.ShipperID = o.ShipVia

WHERE s.CompanyName = 'Federal Shipping';

333 rows returned

**Question 5**

SELECT

c.ContactName

FROM tblCustomers c

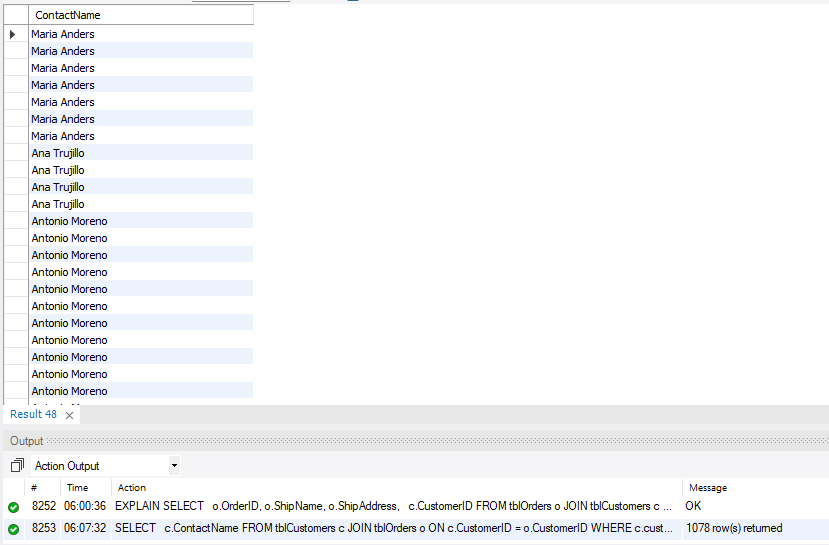
JOIN tblOrders o ON c.CustomerID = o.CustomerID

WHERE c.customerID NOT IN (

SELECT o.customerID FROM tblOrders

WHERE o.OrderDate BETWEEN '2011-01-01 00:00:00' AND '2011-12-31 00:00:00'

);

1078 rows returned

**Question6**

SELECT

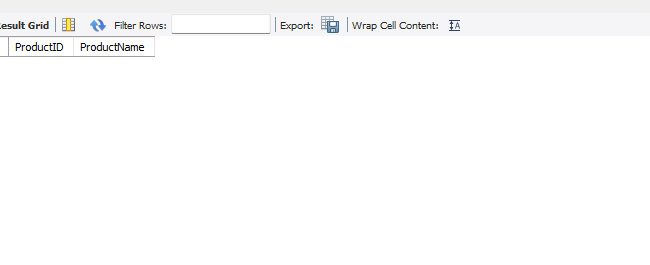
p.ProductID,

p.ProductName

FROM tblProducts p

LEFT JOIN tblOrderDetails od ON p.ProductID = od.ProductID

WHERE od.OrderID IS NULL or od.ProductID IS NULL;

no row returned

**Question 7**

SELECT

c.CustomerID, c.ContactName,

o.OrderID

FROM tblCustomers c

JOIN tblOrders o ON c.CustomerID = o.CustomerID

WHERE c.CompanyName IN

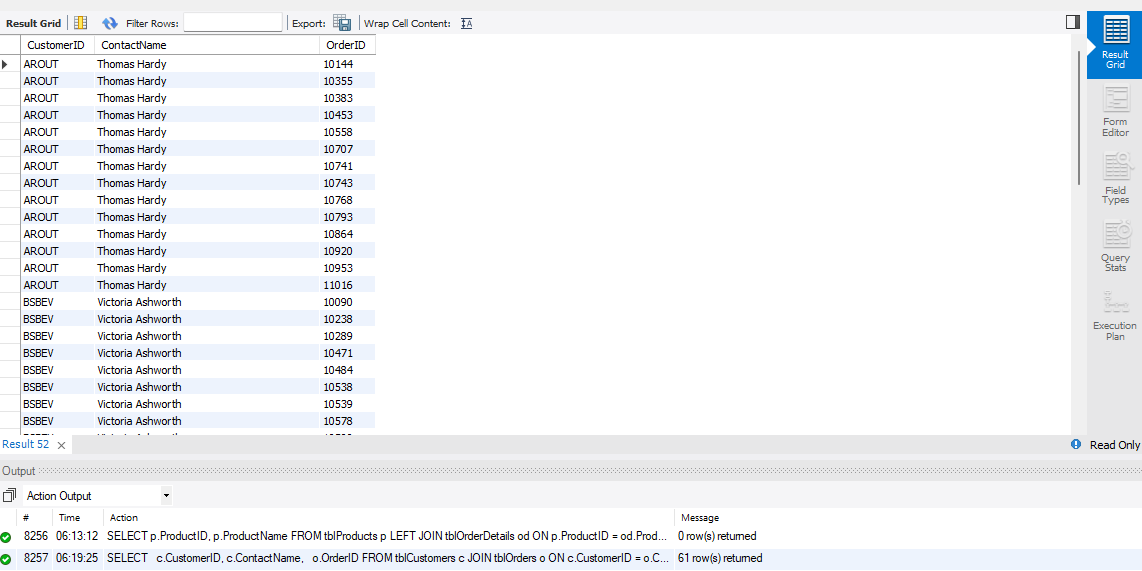
(

SELECT c.CompanyName

FROM tblCustomers c

WHERE c.City = 'London'

);

61 rows returned

**Question 8**

**SELECT**

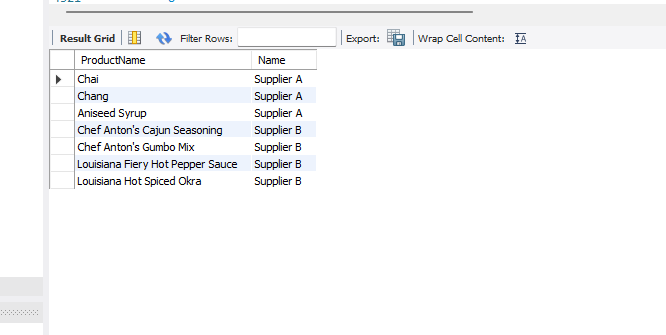
**p.ProductName,**

**s.Name**

**FROM tblProducts p**

**JOIN tblSupplier s ON p.SupplierID = s.SupplierID**

**WHERE s.Name IN ('Supplier A', 'Supplier B');**

****

**Question 9**

SELECT

p.EnglishName,

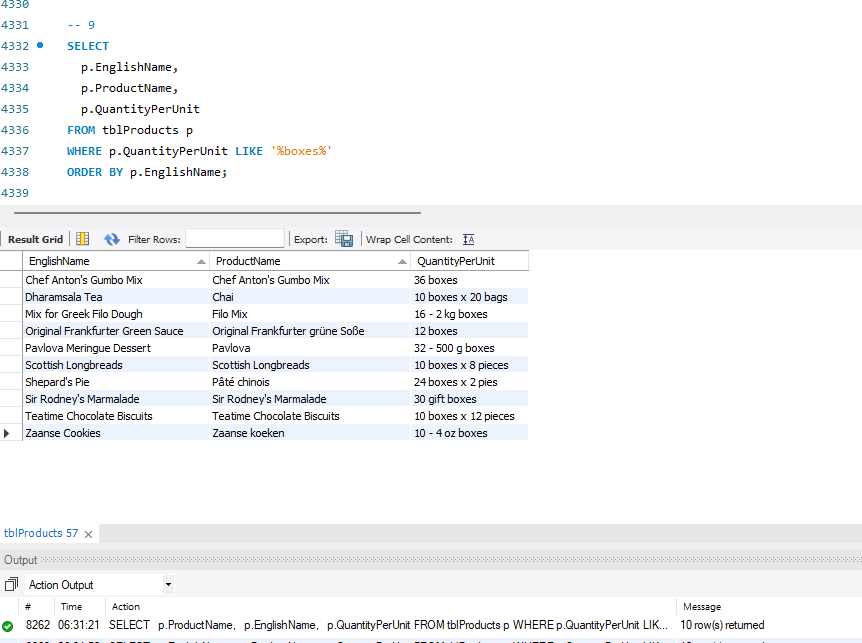
p.ProductName,

p.QuantityPerUnit

FROM tblProducts p

WHERE p.QuantityPerUnit LIKE '%boxes%'

ORDER BY p.EnglishName;

10 rows returned

**Part 3: Update**

**Question 1**

-- create table tblEmployee

use Orders;

CREATE TABLE orders.tblEmployee (

EmployeeID INT UNIQUE NOT NULL AUTO\_INCREMENT,

LastName VARCHAR(30) NOT NULL,

FirstName VARCHAR(30) NOT NULL,

Address VARCHAR(30) NOT NULL,

City VARCHAR(10) NOT NULL,

Province VARCHAR(10) NOT NULL,

Postalcode VARCHAR(6) NOT NULL,

Phone VARCHAR(15) NOT NULL,

Salary FLOAT(12,4) NOT NULL,

PRIMARY KEY (EmployeeID)

);

-- inserting into tblEmployee

INSERT INTO orders.tblEmployee (LastName, FirstName, Address, City, Province, Postalcode, Phone, Salary) VALUES

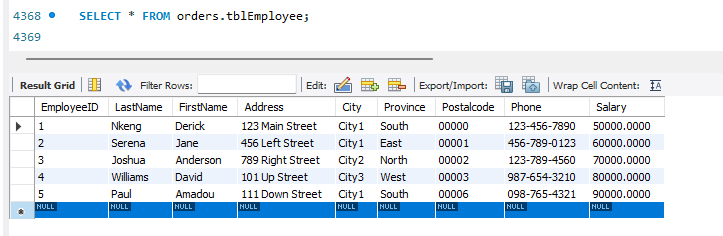
('Nkeng', 'Derick', '123 Main Street', 'City1', 'South', '00000', '123-456-7890', 50000.00),

('Serena', 'Jane', '456 Left Street', 'City1', 'East', '00001', '456-789-0123', 60000.00),

('Joshua', 'Anderson', '789 Right Street', 'City2', 'North', '00002', '123-789-4560', 70000.00),

('Williams', 'David', '101 Up Street', 'City3', 'West', '00003', '987-654-3210', 80000.00),

('Paul', 'Amadou', '111 Down Street', 'City1', 'South', '00006', '098-765-4321', 90000.00);



SELECT \* FROM orders.tblEmployee;

SELECT \* FROM orders.tblEmployee INNER JOIN orders.tblOrders ON tblEmployee.EmployeeID = tblOrders.CustomerID;

**Question 2**

ALTER TABLE orders.tblOrders

ADD TotalSales FLOAT(12,4) NOT NULL;

**Question 3**

UPDATE orders.tblOrders

SET TotalSales = (

SELECT SUM(od.Quantity \* od.UnitPrice \* (1 - od.Discount))

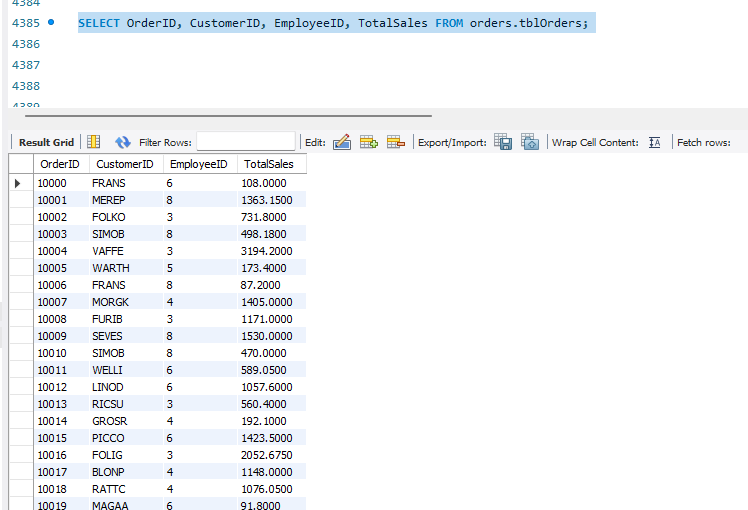
FROM tblOrderDetails od

WHERE od.OrderID = tblOrders.OrderID

GROUP BY OrderID

);

SELECT OrderID, CustomerID, EmployeeID, TotalSales FROM orders.tblOrders;



**CHANGES MADE**

Firstly, the create table statements were modified to integrate primary and foreign keys. The square brackets were removed and go was replaced with “;”, dbo with “orders” and money datatype to float, like wise the nvarchar datatype to varchar. The primary key columns were set to not null as well.

Another measure change was to adjust the table order with respect to the foreign such that tables which were referenced to came before tables referencing to i.e., tables with no foreign key came first before tables which used their primary key as foreign key. The insert into statements were also arranged in this order.

On the insert statements, the N’…’ nvarchar strings were changed to ‘…’ VARCHAR. The date formats were changed too from yyyymmdd HH:MM:SS to yyyy-mm-dd HH:MM:SS with respect to the MySQL DATETIME format. Semicolons were added at the end of each statement

In MySQL, SELECT is the syntax used to print so all print were changed to select

**DIFFICULTIES FACED**

Understanding the complete project as a whole and what was required of me wasn’t easy

Data types: Finding a data type to use in place of money took time to and finally used float. But I learn from online money itself is a datatype. Also changing the date formats to the right one was challenging.

SELECT: Displaying fields from different tables using a single query statement seemed tough.

UPDATE: Joining two tables at the end such that the newly created tblEmployees table has its primary key employeeID as a foreign key in an already existing table tblOrders.

ERRORS: Managing errors was difficult as many errors came at the beginning. The database needed to be dropped to avoid conflicts each time a table wasn’t created or a value wasn’t inserted due to one error or the other. Some errors were strange too

Above all the main challenge was to understand the project. Ones what to be done was understood and everything became easier to implement