|  |  |
| --- | --- |
| **Practicum Case** | Diagram  Description automatically generated |
| COMP6799 | COMP6799001 | COMP6799016 | COMP6799049  Database Technology |
| **Computer Science** | **E231-COMP6799-HG01-05** |
| ***Valid on*** *Even Semester Year 2022/2023* | **Revision 00** |

## Learning Outcomes

* LO2 – explain the concepts of relational algebra, DDL, DML, transaction management on database and concurrency control

## Topic

* Session 05 – Simple Query II

## Sub Topics

* Order By
* Join
* Set Operation
* Review

**Tabel Relasional**

*Relational Table*

Diagram

Description automatically generated

## Soal

*Case*

1. Display **all data** from “**MsCustomer**” table and **sort** it by **CustomerDOB** from the **most recent to the oldest**.

Table

Description automatically generated

select \* from mscustomer order by CustomerDOB desc;

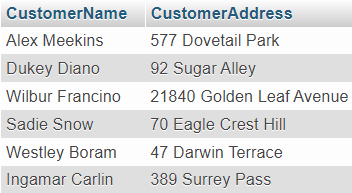
1. Display **ProductName**, **CategoryName**, and **ProductPrice** (obtained by **adding** ‘**$**’ in front of the **ProductPrice**) for every **Dairy** product.

(**CONCAT**)



1. Display **CustomerName** and **CustomerAddress** for every **customer** who have done a transaction served by an **employe with id** “**EM002**” and happened in the **third** **quarter of the year**. Duplicate data must be **shown only once**.

(**DISTINCT**, **QUARTER**)



1. Display **CustomerName**, **CustomerEmail**, **CustomerDOB**, **TransactionID**, and **TransactionDate** for every **customer** whose name **contains letter** “**m**” and is **born in** **January until August** and all of the transaction they have done. If the customer **haven’t done any ransaction** yet, display **NULL** on the transaction column.

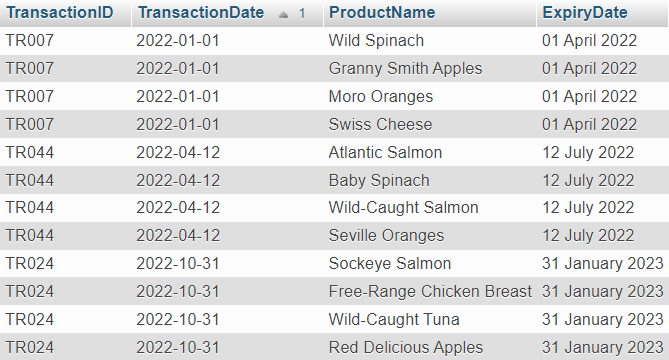
(**LEFT JOIN**, **LIKE**, **MONTH**)

Table

Description automatically generated

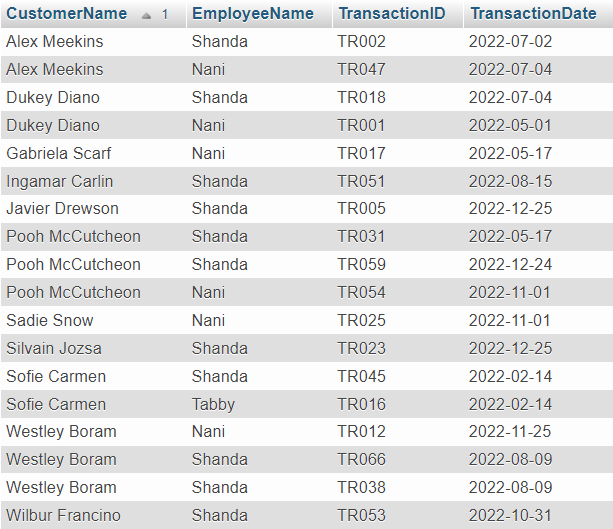
1. Display **TransactionID**, **TransactionDate**, **ProductName**, and **ExpiryDate** (obtained by **adding** **3 months** to the **TransactionDate** and show it in “**dd-mm-yyyy**” **format**) for every **product** bought by a customer whose **email ends with** “**@msu.com**” and **sort** the date from the **oldest to the most recent**.

(**DATE\_FORMAT**, **LIKE**)



1. Display **CustomerName**, **EmployeeName**, **TransactionID**, and **TransactionDate** for every **transaction** which **include** **meat product** in it and the transaction is **served by a male employee**. Then **sort** the data by **CustomerName** in **ascending format**.

(**DISTINCT**, **LIKE**)



1. Display **EmployeeName** (obtained by **adding** “**Mr.**” in front of the **EmployeeName**) and **EmployeeAddress** for every **Male employee** whose **address begin with** **one digit**; then **combine** itwith **EmployeeName** (obtained by **adding** “**Mrs.**” in front of the **EmployeeName**) and **EmployeeAddress** for every **Female employee** whose **address begin with** **one character**.

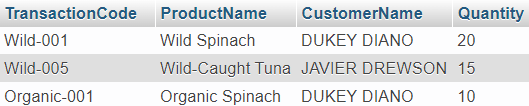
(**CONCAT**, **UNION**, **LIKE**)

Graphical user interface, table

Description automatically generated

1. Display TransactionCode (obtained by replacing “TR” in TransactionID with “Wild-”), ProductName, CustomerName (obtained from CustomerName in uppercase format), and Quantity (obtained from the quantity of product sold) for every product whose name contains “Wild”, is bought by a customer whose name consists of two words, and is bought with quantity more than 12; then combine it with TransactionCode (obtained by replacing “TR” in TransactionID with “Organic-”), ProductName, CustomerName (obtained from CustomerName in uppercase format), and Quantity (obtained from the quantity of product sold) for every product whose name contains “Organic”, is bought by a customer whose name consists of two words, and is bought with quantity less than 12

(**REPLACE**, **UPPER**, **UNION**, **LIKE**)



1. Display **ProductName**, **CategoryName**, and **ProductStock** for a **Dairy product** with the **lowest stock**.

(**LIKE**)

Graphical user interface

Description automatically generated