COMP122 ASSIGNMENT 4

Convenience Store

Group 6

Anmol, Shrestha (301364068)
Drake, Kirkos (301358612)
Xyrylle, Maputol (301362512)
Yik Hei, Chan (30134095)

Table of Contents

1 Introduction

Welcome to our database report. The database we have decided to create is the database for a convenience store brand. When deciding on a topic for our report, we thought of many different databases we could potentially create. Eventually, we decided on a database for a convenience store. The decision to create a database for a convenience store was due to the great potential, and vast amounts of data which could be present within a database for a convenience store brand or any other retail chain, which made it a very compelling database to create.

1.1 Overview of Report

The Overview/layout of our report is as follows:

- Data Requirements:

  The data requirements section of our report explains how we gathered our data, used our data, the main objectives/goals, and the required data needed for our database.

- ERD Modeling diagram:
  The ERD model shows each entity within our database, along with their attributes, and how they and the attributes connect to each other.

- Table descriptions/information:
  This part of the database report visually shows and explains each table. This includes the Table name, The different attributes within the table, and other information present within the table.

- insert & query:
  The insert and query section of our report shows the queries created for the database along with insert.

1.2 Purpose of Project

1. Our report has many purposes for our project. Firstly, our report will serve as a tool to record our progress, ideas, and the database creation process. This will allow us to potentially look back for future reference if required. By achieving this, our report can be used to check milestones through out our project and see if any deviations occur, and how we may fix it if required.

2. Our report will allow us to both demonstrate and enhance our knowledge on database creation, and design. Additionally, it will allow us to develop problem solving skills related to the creation and designing of various databases. Adding on, this will further our knowledge on collaboration and communication within a group setting.

3. This report will serve as a useful tool for reflection on our skills. For example, this report will highlight any weakness we may suffer from and allow us to correct them by reflecting on these issues.

what does our database provide:

Our system provides efficiency at a fast pace. For example, shipments, sales, and scheduling will all have improved speed, and efficiency to prevent issues/errors which many occur. Additionally, our database can help streamline inventory/stock of products, employee scheduling, and customer loyalty rewards.

2 Data Requirement Phase

Goals/Requirements

For our database report we first had to produce a set of requirements and goals that were necessary for our database to be successful. Firstly, for many convenience/retail stores it is vital to have a well-functioning database. This is due to the huge amounts of data needed to run a convenience store. For example, a convenience store will contain data such as shipment orders, customer shopping trends, staff information, and many other vital components necessary to function properly and successfully. Due to convenience stores, or any type of retail store having many quick moving parts (such as orders, loyal customers, etc.), our database must be both very efficient and

functional. This is to prevent slowdowns and various other issues which may occur within an inefficient database. To accomplish this, we must design a database with a well-thought-out layout to allow for efficient retrieval and storage.

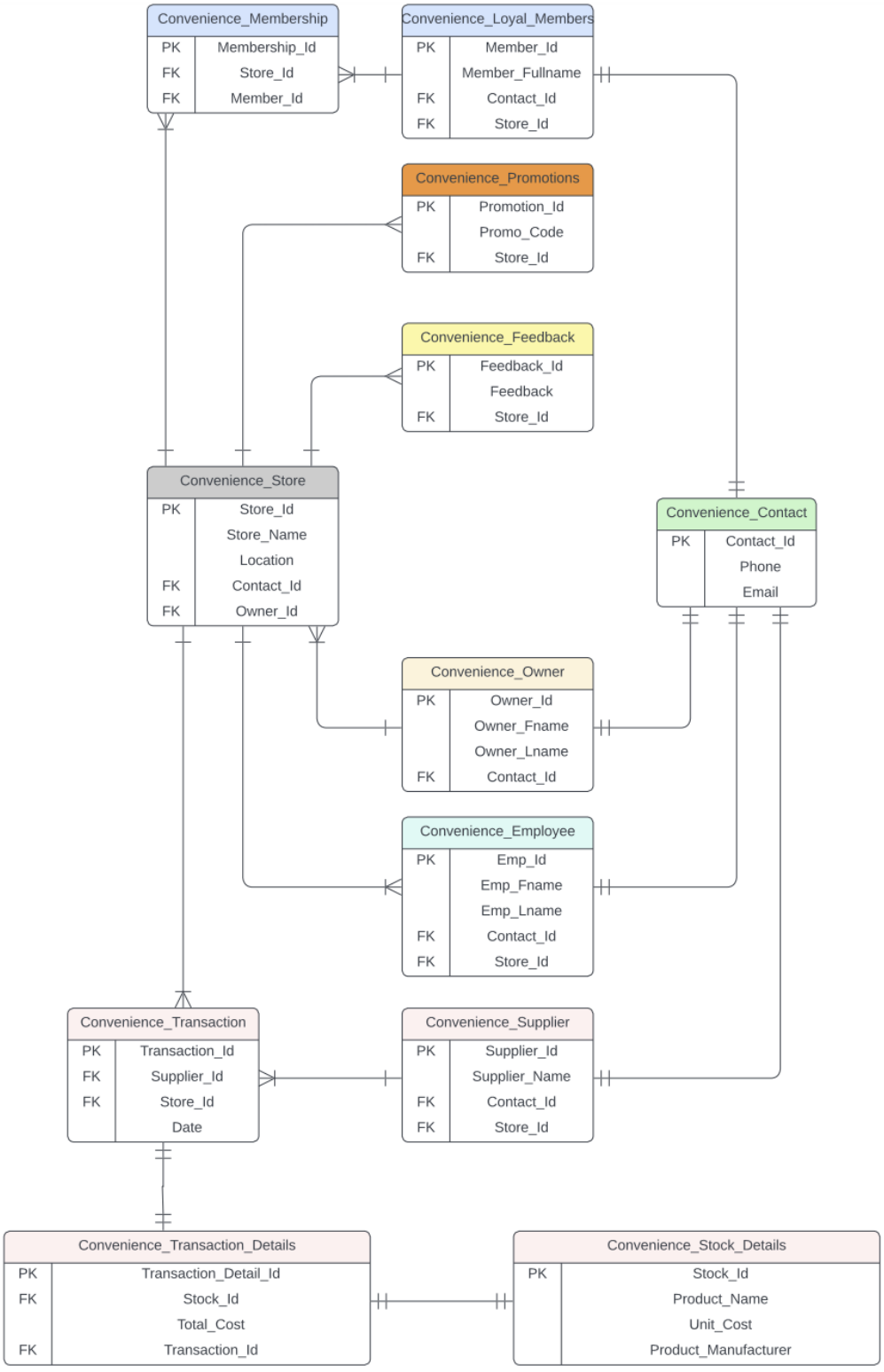
To satisfy the needs of the convenience store brand we need many key entities to satisfy and fulfill the many parts needed to run a retail/convenience brand properly and effectively, with the main goal of having a well-structured/functional database. If we fail to fully plan each entity, their attributes and connections between them, our database will be messy and inefficient, causing many problems in the future, or present moment. Additionally, our database must be future proof, meaning if anything changes in the future, the database will easily accommodate these changes. For example, our database must be able to accommodate things such as regulatory changes, various market changes, business growth/expansion, and many other changes that could occur.

Required Data and how we gathered it.

There are many different types of data required for our database to function correctly, and efficiently for a convenience store. Firstly, we need data on employees and owners. This data will include Full name, ID, Contact ID, and store ID specifically for employees. Secondly, we need data on customers. Customer data includes membership, and customer loyalty. Thirdly, we need data on the convenience store itself, such as the location of the store, store id, name, contact id and the owner ID. Fourth, we need data on feedback from customers, which includes ID, the feedback, and the store ID. Fifth, we need data on store promotions. This includes data on promotion ID's, promo codes, and the store ID. Sixth, we need data on the supplier, such as the ID, name, contact ID, and store ID. Additionally we need 3 more sets of data on transactions within the store, transaction details, and stock details.

We gathered and decided on this data through two different methods. Firstly, we utilized google to search and a few retail/convenience store websites to work out some potential data which may be stored inside of a convenience store database. Furthermore, in addition to google, we utilized various brainstorming techniques as a team to also find potential data which may be stored inside of a convenience store database.

3 ERD


Convenience_Membership
PK Membership_Id
FK Store_Id
FK Member_Id
Convenience_Loyal_Members
PK Member_Id
Member_Fullname
FK Contact_Id
FK Store_Id
Convenience_Promotions
PK Promotion_Id
Promo_Code
FK Store_Id
Convenience_Feedback
PK Feedback_Id
Feedback
FK Store_Id
Convenience_Store
PK Store_Id
Store_Name
Location
FK Contact_Id
FK Owner_Id
Convenience_Contact
PK Contact_Id
Phone
Email
Convenience_Owner
PK Owner_Id
Owner_Fname
Owner_Lname
FK Contact_Id
Convenience_Employee
PK Emp_Id
Emp_Fname
Emp_Lname
FK Contact_Id
FK Store_Id
Convenience_Transaction
PK Transaction_Id
FK Supplier_Id
FK Store_Id
Date
Convenience_Supplier
PK Supplier_Id
Supplier_Name
FK Contact_Id
FK Store_Id
Convenience_Transaction_Details
PK Transaction_Detail_Id
FK Stock_Id
Total_Cost
FK Transaction_Id
Convenience_Stock_Details
PK Stock_Id
Product_Name
Unit_Cost
Product_Manufacturer

4 Table Description

This tables includes information of the convenience store's membership which includes:

Field Name	Datatype (length)	Constraints	Constraints Name
Membership_Id	NUMBER(10)	Primary Key	N/A
Store_Id	NUMBER(10)	Foreign Key	FK_Membership_Store
Member_Id	NUMBER(10)	Foreign Key	FK_Membership_Member

This table consists of information about the store's loyal members.

Field Name	Datatype (length)	Constraints	Constraints Name
Member_ID	NUMBER(10)	Primary Key	N/A
Member_Fullname	VARCHAR2(50)	Not null	N/A
Contact_Id	Number(10)	Foreign key	FK_LoyalCustomer_Contact
Store_Id	Number(10)	Foreign key	FK_LoyalCustomer_Store

This table holds data about the store's ongoing promotions which includes:

Field Name	Datatype (length)	Constraints	Constraints Name
Promotion_Id	NUMBER(10)	Primary Key	N/A
Promo_Code	VARCHAR2(10)	Not null	N/A
Store_Id	NUMBER(10)	Foreign Key	FK_Promototion_Store

This table stores customer feedback data about the convenience store.

Field Name	Datatype (length)	Constraints	Constraints Name
Feedback_Id	NUMBER(10)	Primary key	N/A
Feedback	VARCHAR2(255)	Not null	n/a
Store_Id	Number(10)	Foreign Key	FK_Feedback_Store

This table lists information regarding the conveinece store.

Field Name	Datatype (length)	Constraints	Constraints Name

Store_Id	NUMBER(10)	Primary Key	N/A
Store_Name	VARCHAR2(50)	Not null	N/A
Location	VARCHAR2(50)	Not null	N/A
Contact_Id	NUMBER(10)	Foreign Key	FK_Store_Contact
Owner_Id	NUMBER(10)	Foreign Key	FK_Store_Owner

This table has the convenience store's owner's details:

Field Name	Datatype (length)	Constraints	Constraints Name
Owner_Id	NUMBER(10)	Primary key	N/A
Owner_Fname	VARCHAR2(20)	Not null	N/A
Owner_Lname	VARCHAR2(20)	Not null	N/A
Contact_Id	NUMBER(10)	Foreign key	FK_Owner_Contact

This table is about the store's employees with the following details:

Field Name	Datatype (length)	Constraints	Constraints Name
Emp_Id	NUMBER(10)	Primary key	N/A
Emp_Fname	Varchar2(20)	Not null	N/A
Emp_Lname	Varchar2(20)	Not null	N/A
Contact_Id	NUMBER(10)	Foreign key	FK_Employee_Contact
Store_Id	NUMBER(10)	Foreign key	FK_Employee_Store

This table is about any transaction made by the store. It consists of:

Field Name	Datatype (length)	Constraints	Constraints Name
Transaction_Id	NUMBER(10)	Primary key	N/A
Supplier_Id	NUMBER(10)	Foreign Key	FK_Transaction_Supplier
Store_Id	NUMBER(10)	Foreign key	FK_Transaction_Store
Date	DATE	Not null	N/A

This table holds information of the convenience store's supplier(s) which has information of:

Field Name	Datatype (length)	Constraints	Constraints Name

Supplier_ID	NUMBER(10)	Primary key	N/A
Supplier_Name	VARCHAR2(50)	Not null	N/A
Contact_Id	NUMBER(10)	Foreign key	FK_Supplier_Contact
Store_Id	NUMBER(10)	Foreign key	FK_Supplier_Store

This table lists the store's contact information. It has:

Field Name	Datatype (length)	Constraints	Constraints Name
Contact_Id	NUMBER(10)	Primary key	N/A
Phone	VARCHAR2(20)	Not null	N/A
Email	VARCHAR2(50)	Not null	N/A

This table holds the details of the transactions made. It includes:

Field Name	Datatype (length)	Constraints	Constraints Name
Transaction_Detail_Id	NUMBER(10)	Primary key	N/A
Stock_Id	NUMBER(10)	Foreign key	FK_Stock_Id
Total_Cost	NUMBER(10)	Not null	N/A
Transaction_Id	Number(10)	Foreign key	FK_Transaction_Id

This table holds information of the store's stock details. It has:

Field Name	Datatype (length)	Constraints	Constraints Name
Store_Id	NUMBER(10)	Primary key	N/A
Product_Name	VARCHAR2(50)	Not null	N/A
Unit_Cost	NUMBER(10,2)	Not null	N/A
Product_Manufacturer	VARCHAR2(50)	Not null	N/A

5 Table Creation & Data Insert

5.1 Contact

```
CREATE TABLE Contact (

    PKC_Id NUMBER(10) PRIMARY KEY,

    Phone VARCHAR2(20),

    Email VARCHAR2(255)

);
```

```
INSERT INTO Contact
VALUES (1001, '613-555-0164', 'nop-icisohi63@hotmail.com');
INSERT INTO Contact
VALUES (1002, '613-555-0156', 'hijofol-owe73@mail.com');
INSERT INTO Contact
VALUES (1003, '613-555-0119', 'joruca_yidu87@hotmail.com');
INSERT INTO Contact
VALUES (1004, '613-555-0180', 'dovupoh_epi93@yahoo.com');
INSERT INTO Contact
VALUES (1005, '613-555-0140', 'bun-ibubize75@yahoo.com');
INSERT INTO Contact
VALUES (1006, '613-555-0147', 'riwe-xijojo95@gmail.com');
INSERT INTO Contact
VALUES (1007, '647-969-8033', 'mos_okumexi97@hotmail.com');
INSERT INTO Contact
VALUES (1008, '613-555-0172', 'wisure_joye16@hotmail.com');
INSERT INTO Contact
VALUES (1009, '613-555-0129', 'kubuxic-ozo80@aol.com');
INSERT INTO Contact
VALUES (1010, '613-555-0124', 'rigire_wika97@hotmail.com');
INSERT INTO Contact
VALUES (1011, '613-555-0153', 'pesu_lifame4@gmail.com');
```

	PKC_ID	PHONE	EMAIL
1	1001	613-555-0164	nop-icisohi63@hotmail.com
2	1002	613-555-0156	hijofol-owe73@mail.com
3	1003	613-555-0119	joruca_yidu87@hotmail.com
4	1004	613-555-0180	dovupoh_epi93@yahoo.com
5	1005	613-555-0140	bun-ibubize75@yahoo.com
6	1006	613-555-0147	riwe-xijojo95@gmail.com
7	1007	647-969-8033	mos_okumexi97@hotmail.com
8	1008	613-555-0172	wisure_joye16@hotmail.com
9	1009	613-555-0129	kubuxic-ozo80@aol.com
10	1010	613-555-0124	rigire_wika97@hotmail.com
11	1011	613-555-0153	pesu_lifame4@gmail.com

5.2 Owner

```
CREATE TABLE Owner (
    PKO_Id NUMBER(10) PRIMARY KEY,
    OFname VARCHAR2(50),
    OLname VARCHAR2(50),
    FKC_Id NUMBER(10),
    CONSTRAINT FK_Owner_Contact FOREIGN KEY (FKC_Id) REFERENCES Contact(PKC_Id)
);
```

```
INSERT INTO Owner
VALUES (1001, 'Beau', 'Young', 1001);
INSERT INTO Owner
VALUES (1002, 'Hunter', 'March', 1002);
INSERT INTO Owner
VALUES (1003, 'Malcolm', 'Kelly', 1003);
INSERT INTO Owner
VALUES (1004, 'Rory', 'Reeves', 1004);
INSERT INTO Owner
VALUES (1005, 'Tobias', 'Clarkson', 1005);
```

	PKO_ID	OFNAME	OLNAME	FKC_ID
1	1001	Beau	Young	1001
2	1002	Hunter	March	1002
3	1003	Malcolm	Kelly	1003
4	1004	Rory	Reeves	1004
5	1005	Tobias	Clarkson	1005

5.3 Store

```
CREATE TABLE Store (

    PKS_Id NUMBER(10) PRIMARY KEY,

    SName VARCHAR2(255),

    Location VARCHAR2(255),

    FKC_Id NUMBER(10),

    FKO_Id NUMBER(10),

    CONSTRAINT FK_Store_Contact FOREIGN KEY (FKC_Id) REFERENCES Contact(PKC_Id),

    CONSTRAINT FK_Store_Owner FOREIGN KEY (FKO_Id) REFERENCES Owner(PKO_Id)

);
```

```
INSERT INTO Store
VALUES (1001, '3mile Convenience Store', 'CA ON Toronto, 70 Town Centre Court unit 4, M1P 0B2', 1007, 1001);
INSERT INTO Store
VALUES (1002, 'Toronto Convenience', '267 Morningside Ave, Scarborough, ON M1E 3G1', 1002, 1002);
INSERT INTO Store
VALUES (1003, 'The Corner Convenience', '393 King St E unit1, Toronto, ON M5A 1L3', 1003, 1003);
INSERT INTO Store
VALUES (1004, 'Joy Convenience Store', '736 Bay St., Toronto, ON M5G 2M4', 1004, 1004);
INSERT INTO Store
VALUES (1005, 'Circle K', '10 King St E, Toronto, ON M5C 1C3', 1005, 1005);
```

	PKS_ID	SNAME	LOCATION	FKC_ID	FKO_ID
1	1001	3mile Convenience Store	CA ON Toronto, 70 Town Centre Court unit 4, M1P 0B2	1007	1001
2	1002	Toronto Convenience	267 Morningside Ave, Scarborough, ON M1E 3G1	1002	1002
3	1003	The Corner Convenience	393 King St E unit1, Toronto, ON M5A 1L3	1003	1003
4	1004	Joy Convenience Store	736 Bay St., Toronto, ON M5G 2M4	1004	1004
5	1005	Circle K	10 King St E, Toronto, ON M5C 1C3	1005	1005

5.4 Employee

```
CREATE TABLE Employee (

    PKEmp_Id NUMBER(10) PRIMARY KEY,

    EFname VARCHAR2(50),

    ELname VARCHAR2(50),

    FKC_Id NUMBER(10),

    FKS_Id NUMBER(10),

    CONSTRAINT FK_Employee_Contact FOREIGN KEY (FKC_Id) REFERENCES Contact(PKC_Id),

    CONSTRAINT FK_Employee_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)

);
```

```
INSERT INTO Employee
VALUES (1001, 'Jude', 'Brown', 1006, 1001);
INSERT INTO Employee
VALUES (1002, 'Marcus', 'Markle', 1008, 1005);
INSERT INTO Employee
VALUES (1003, 'Ashton', 'Park', 1009, 1004);
INSERT INTO Employee
VALUES (1004, 'Frank', 'Tyrell', 1010, 1005);
INSERT INTO Employee
VALUES (1005, 'Hugo', 'Symons', 1011, 1002);
```

	PK...	EFNA...	ELNAME	FKC_ID	FKS_ID
1	1001	Jude	Brown	1006	1001
2	1002	Marcus	Markle	1008	1005
3	1003	Ashton	Park	1009	1004
4	1004	Frank	Tyrell	1010	1005
5	1005	Hugo	Symons	1011	1002

5.5 Supplier

```
CREATE TABLE Supplier (
    PKSupp_Id NUMBER(10) PRIMARY KEY,
    Supp_Name VARCHAR2(255),
    FKC_Id NUMBER(10),
    FKS_Id NUMBER(10),
    CONSTRAINT FK_Supplier_Contact FOREIGN KEY (FKC_Id) REFERENCES Contact(PKC_Id),
    CONSTRAINT FK_Supplier_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)
);
```

```
INSERT INTO Supplier
VALUES (1001, 'Faire', 1006, 1001);
INSERT INTO Supplier
VALUES (1002, 'Canpipe', 1008, 1003);
INSERT INTO Supplier
VALUES (1003, 'Canova', 1009, 1004);
INSERT INTO Supplier
VALUES (1004, 'Carson Supply', 1010, 1005);
INSERT INTO Supplier
VALUES (1005, 'Insurge Inc.', 1011, 1002);
```

	PKSUPP_ID	SUPP_NAME	FKC_ID	FKS_ID
1	1001	Faire	1006	1001
2	1002	Canpipe	1008	1003
3	1003	Canova	1009	1004
4	1004	Carson Supply	1010	1005
5	1005	Insurge Inc.	1011	1002

5.6 Feedback

```
CREATE TABLE Feedback (

    PKFeed_Id NUMBER(10) PRIMARY KEY,

    Feedback VARCHAR2(255),

    FKS_Id NUMBER(10),

    CONSTRAINT FK_Feedback_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)

);
```

```
INSERT INTO Feedback
VALUES (1001, 'Great service!', 1001);
INSERT INTO Feedback
VALUES (1002, 'Clean store.', 1003);
INSERT INTO Feedback
VALUES (1003, 'Friendly staff.', 1004);
INSERT INTO Feedback
VALUES (1004, 'Product variety could be better.', 1005);
INSERT INTO Feedback
VALUES (1005, 'Quick checkout process.', 1002);
```

	PKFEED_ID	FEEDBACK	FKS_ID
1	1001	Great service!	1001
2	1002	Clean store.	1003
3	1003	Friendly staff.	1004
4	1004	Product variety could be better.	1005
5	1005	Quick checkout process.	1002

5.7 Promotions

```
CREATE TABLE Promotions (
    PKPromo_Id NUMBER(10) PRIMARY KEY,
    Promo_Code VARCHAR2(50),
    FKS_Id NUMBER(10),
    CONSTRAINT FK_Promotions_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)
);
```

```
INSERT INTO Promotions
VALUES (1001, 'SAVE10', 1001);
INSERT INTO Promotions
VALUES (1002, 'SPRING20', 1003);
INSERT INTO Promotions
VALUES (1003, 'SUMMER25', 1004);
INSERT INTO Promotions
VALUES (1004, 'FALL30', 1005);
INSERT INTO Promotions
VALUES (1005, 'WINTER15', 1002);
```

	PKPROMO_ID	PROMO_CODE	FKS_ID
1	1001	SAVE10	1001
2	1002	SPRING20	1003
3	1003	SUMMER25	1004
4	1004	FALL30	1005
5	1005	WINTER15	1002

5.8 Loyal_Members

```
CREATE TABLE Loyal_Members (
    PKMem_Id NUMBER(10) PRIMARY KEY,
    Mem_Fullname VARCHAR2(255),
    FKC_Id NUMBER(10),
    FKS_Id NUMBER(10),
    CONSTRAINT FK_Member_Contact FOREIGN KEY (FKC_Id) REFERENCES Contact(PKC_Id),
    CONSTRAINT FK_Member_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)
);
```

```
INSERT INTO Loyal_Members
VALUES (1001, 'Patrick Banks', 1002, 1001);
INSERT INTO Loyal_Members
VALUES (1002, 'Lawson Gosling', 1004, 1001);
INSERT INTO Loyal_Members
VALUES (1003, 'Dawson McLeod', 1001, 1004);
INSERT INTO Loyal_Members
VALUES (1004, 'Jeremy Piers', 1003, 1005);
INSERT INTO Loyal_Members
VALUES (1005, 'Jonathan Irvin', 1010, 1002);
```

	PKMEM_ID	MEM_FULLNAME	FKC_ID	FKS_ID
1	1001	Patrick Banks	1002	1001
2	1002	Lawson Gosling	1004	1001
3	1003	Dawson McLeod	1001	1004
4	1004	Jeremy Piers	1003	1005
5	1005	Jonathan Irvin	1010	1002

5.9 Membership

```
CREATE TABLE Membership (
    PKMemb_Id NUMBER(10) PRIMARY KEY,
    FKS_Id NUMBER(10),
    FKMem_Id NUMBER(10),
    CONSTRAINT FK_Membership_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id),
    CONSTRAINT FK_Membership_Member FOREIGN KEY (FKMem_Id) REFERENCES Loyal_Members(PKMem_Id)
);
```

```
INSERT INTO Membership
VALUES (1001, 1001, 1002);
INSERT INTO Membership
VALUES (1002, 1004, 1003);
INSERT INTO Membership
VALUES (1003, 1005, 1004);
INSERT INTO Membership
VALUES (1004, 1002, 1005);
INSERT INTO Membership
VALUES (1005, 1001, 1001);
```

	PKMEMB_ID	FKS_ID	FKMEM_ID
1	1001	1001	1002
2	1002	1004	1003
3	1003	1005	1004
4	1004	1002	1005
5	1005	1001	1001

5.10 Stock_Details

```
CREATE TABLE Stock_Details (

    PKStock_Id NUMBER(10) PRIMARY KEY,

    PName VARCHAR2(255),

    Unit_Cost NUMBER(10,2),

    Manufacturer VARCHAR2(255)

);
```

```
INSERT INTO Stock_Details
VALUES (1001, 'Bread', 2.50, 'ABC Bakery');
INSERT INTO Stock_Details
VALUES (1002, 'Milk', 1.99, 'XYZ Farms');
INSERT INTO Stock_Details
VALUES (1003, 'Bottled Water', 1.00, 'Aqua Co.');
INSERT INTO Stock_Details
VALUES (1004, 'Candy Bars', 1.25, 'Sweet Treats Inc.');
INSERT INTO Stock_Details
VALUES (1005, 'Chips', 2.50, 'Snack Brands');
```

	PKSTOCK_ID	PNAME	UNIT_COST	MANUFACTURER
1	1001	Bread	2.5	ABC Bakery
2	1002	Milk	1.99	XYZ Farms
3	1003	Bottled Water	1	Aqua Co.
4	1004	Candy Bars	1.25	Sweet Treats Inc.
5	1005	Chips	2.5	Snack Brands

5.11 Transaction

```
CREATE TABLE Transaction (
    PKTrans_Id NUMBER(10) PRIMARY KEY,
    FKSupp_Id NUMBER(10),
    FKS_Id NUMBER(10),
    Trans_Date DATE,
    CONSTRAINT FK_Transaction_Supplier FOREIGN KEY (FKSupp_Id) REFERENCES Supplier(PKSupp_Id),
    CONSTRAINT FK_Transaction_Store FOREIGN KEY (FKS_Id) REFERENCES Store(PKS_Id)
);
```

```
INSERT INTO Transaction
VALUES (1001, 1005, 1002, TO_DATE('2024-04-04', 'YYYY-MM-DD'));
INSERT INTO Transaction
VALUES (1002, 1004, 1002, TO_DATE('2024-04-04', 'YYYY-MM-DD'));
INSERT INTO Transaction
VALUES (1003, 1001, 1005, TO_DATE('2024-04-06', 'YYYY-MM-DD'));
INSERT INTO Transaction
VALUES (1004, 1001, 1004, TO_DATE('2024-04-05', 'YYYY-MM-DD'));
INSERT INTO Transaction
VALUES (1005, 1001, 1002, TO_DATE('2024-03-20', 'YYYY-MM-DD'));
```

	PKTRANS_ID	FKSUPP_ID	FKS_ID	TRANS_DATE
1	1001	1005	1002	04-APR-24
2	1002	1004	1002	04-APR-24
3	1003	1001	1005	06-APR-24
4	1004	1001	1004	05-APR-24
5	1005	1001	1002	20-MAR-24

5.12 Transaction_Details

```
CREATE TABLE Transaction_Details (
    PKTransDet_Id NUMBER(10) PRIMARY KEY,
    FKStock_Id NUMBER(10),
    Total_Cost NUMBER(10,2),
    FKTrans_Id NUMBER(10),
    CONSTRAINT FK_Transaction_Detail_Stock FOREIGN KEY (FKStock_Id) REFERENCES Stock_Details(PKStock_Id),
    CONSTRAINT FK_Transaction_Detail_Transaction FOREIGN KEY (FKTrans_Id) REFERENCES Transaction(PKTrans_Id)
);
```

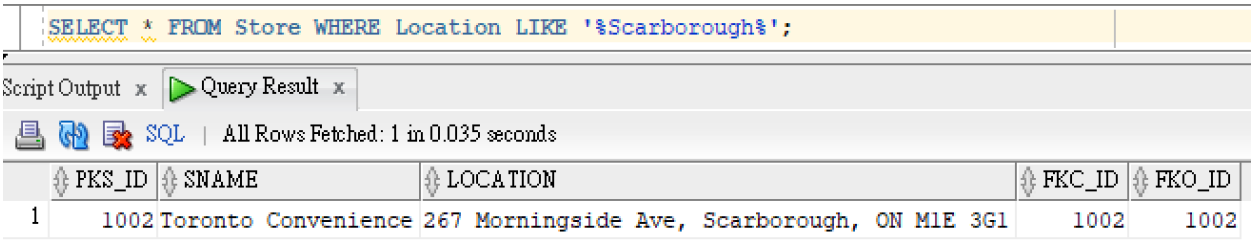
```
INSERT INTO Transaction_Details
VALUES (1001, 1002, 99.5, 1001);
INSERT INTO Transaction_Details
VALUES (1002, 1004, 250, 1002);
INSERT INTO Transaction_Details
VALUES (1003, 1001, 250, 1005);
INSERT INTO Transaction_Details
VALUES (1004, 1003, 100, 1004);
INSERT INTO Transaction_Details
VALUES (1005, 1003, 150, 1003);
```

	PKTRANSDET_ID	FKSTOCK_ID	TOTAL_COST	FKTRANS_ID
1	1001	1002	99.5	1001
2	1002	1004	250	1002
3	1003	1001	250	1005
4	1004	1003	100	1004
5	1005	1003	150	1003

6 Query Results

1. Find the stores that located at Scarborough.

SELECT * FROM Store WHERE Location LIKE '%Scarborough%';



```
SELECT * FROM Store WHERE Location LIKE '%Scarborough%';
```

Script Output x | Query Result x

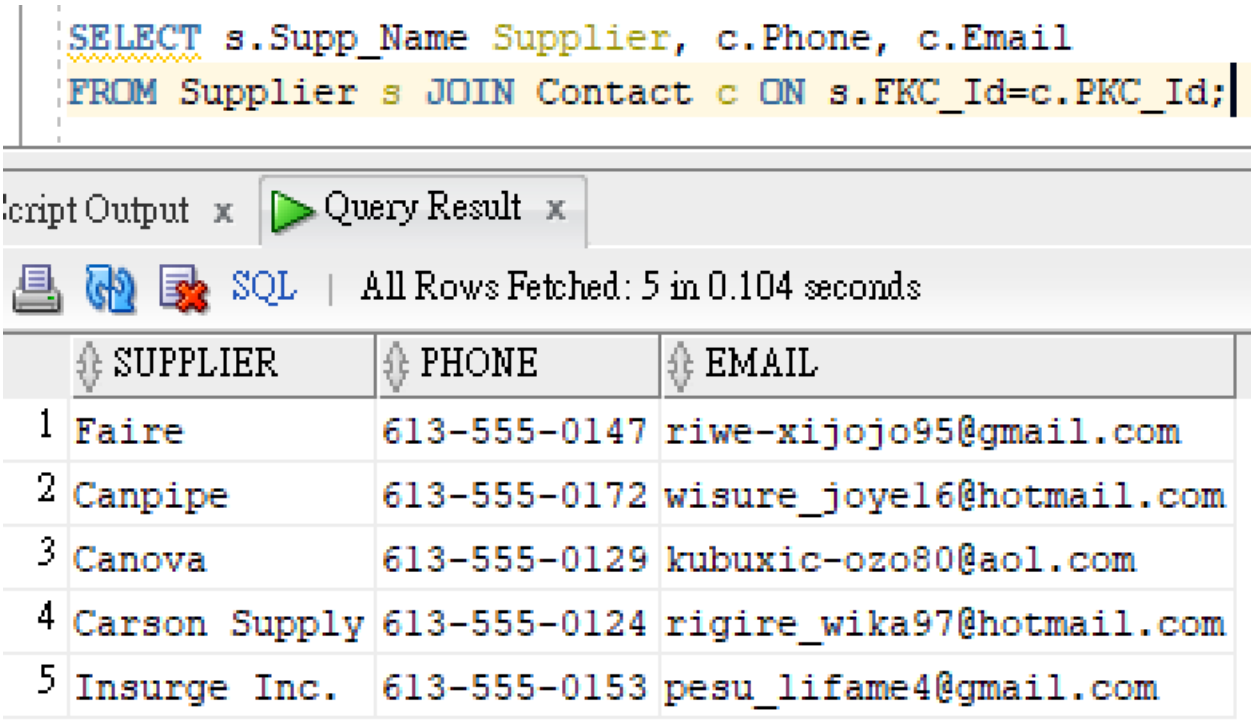
SQL | All Rows Fetched: 1 in 0.035 seconds

	PKS_ID	SNAME	LOCATION	FKC_ID	FKO_ID
1	1002	Toronto Convenience	267 Morningside Ave, Scarborough, ON M1E 3G1	1002	1002

2. Generate the Contact List of suppliers

SELECT s.Supp_Name Supplier, c.Phone, c.Email

FROM Supplier s JOIN Contact c ON s.FKC_Id=c.PKC_Id;



```
SELECT s.Supp_Name Supplier, c.Phone, c.Email
FROM Supplier s JOIN Contact c ON s.FKC_Id=c.PKC_Id;
```

Script Output x | Query Result x

SQL | All Rows Fetched: 5 in 0.104 seconds

	SUPPLIER	PHONE	EMAIL
1	Faire	613-555-0147	riwe-xijojo95@gmail.com
2	Canpipe	613-555-0172	wisure_joye16@hotmail.com
3	Canova	613-555-0129	kubuxic-ozo80@aol.com
4	Carson Supply	613-555-0124	rigire_wika97@hotmail.com
5	Insurge Inc.	613-555-0153	pesu_lifame4@gmail.com

3. Generate the Contact List of owners

SELECT s.SName Store, o.OFname || ', ' || o.OLname "Owner Name", c.Phone

FROM Store s JOIN Owner o ON s.FKO_Id=o.PKO_Id

JOIN Contact c ON o.FKC_Id=c.PKC_Id;

	STORE	Owner Name	PHONE
1	3mile Convenience Store	Beau, Young	613-555-0164
2	Toronto Convenience	Hunter, March	613-555-0156
3	The Corner Convenience	Malcolm, Kelly	613-555-0119
4	Joy Convenience Store	Rory, Reeves	613-555-0180
5	Circle K	Tobias, Clarkson	613-555-0140

4. Generate the Contact List of Loyal Members in "3mile Convenience Store"

```
SELECT l.Mem_Fullname, c.Phone, c.Email
FROM Store s JOIN Loyal_Members l ON s.PKS_Id=l.FKS_Id
JOIN Contact c ON c.PKC_Id=l.FKC_Id
WHERE s.PKS_Id=1001;
```

	MEM_FULLNAME	PHONE	EMAIL
1	Patrick Banks	613-555-0156	hijofol-owe73@mail.com
2	Lawson Gosling	613-555-0180	dovupoh_epi93@yahoo.com

5. Generate the Contact List of all employees working in "Circle K".

```
SELECT e.EFname || ', ' || e.ELname Employee, c.Phone, c.Email
FROM Store s JOIN Employee e ON e.FKS_Id=s.PKS_Id
JOIN Contact c ON c.PKC_Id=e.FKC_Id
WHERE s.PKS_Id=1005;
```

	EMPLOYEE	PHONE	EMAIL
1	Marcus, Markle	613-555-0172	wisure_joye16@hotmail.com
2	Frank, Tyrell	613-555-0124	rigire_wika97@hotmail.com

6. Find the information about each product in stock

```
SELECT sd.PName Product, sd.Unit_Cost,
TO_CHAR(SUM(td.Total_Cost), '$999.9') "Total Cost Spent"
```

FROM Stock_Details sd JOIN Transaction_Details td ON sd.PKStock_Id=td.FKStock_Id

GROUP BY sd.PName, sd.Unit_Cost;

	PRODUCT	UNIT_COST	Total Cost Spent
1	Bottled Water	1	$250.0
2	Milk	1.99	$99.5
3	Bread	2.5	$250.0
4	Candy Bars	1.25	$250.0

7. List all transactions made with "Faire".

SELECT t.PKTrans_Id, t.Trans_Date, td.Total_Cost

FROM Supplier s JOIN Transaction t ON s.PKSupp_Id=t.FKSupp_Id

JOIN Transaction_Details td ON t.PKTrans_Id=td.FKTrans_Id

WHERE s.PKSupp_Id=1001;

	PKTRANS_ID	TRANS_DATE	TOTAL_COST
1	1005	20-MAR-24	250
2	1004	05-APR-24	100
3	1003	06-APR-24	150

8. List all transactions made before April 2024.

SELECT s.SName Store, sd.PName Product, td.Total_Cost, t.Trans_Date

FROM Transaction t JOIN Transaction_Details td ON t.PKTrans_Id=td.FKTrans_Id

JOIN Stock_Details sd ON td.FKStock_Id=sd.PKStock_Id

JOIN Store s ON t.FKS_Id=s.PKS_Id

WHERE t.Trans_Date<='31-MAR-24';

	STORE	PRODUCT	TOTAL_COST	TRANS_DATE
1	Toronto Convenience	Bread	250	20-MAR-24

9. Find the location, owner, feedback, and promotion code of "The Corner Convenience".

```
SELECT s.SName Store, s.Location,
o.OFname || ', ' || o.OLname Owner, f.Feedback, p.Promo_Code
FROM Store s JOIN Owner o ON s.FKO_Id=o.PKO_Id
JOIN Feedback f ON s.PKS_Id=f.FKS_Id
JOIN Promotions p ON s.PKS_Id=p.FKS_Id
WHERE s.PKS_Id=1003;
```

	STORE	LOCATION	OWNER	FEEDBACK	PROMO_CODE
1	The Corner Convenience	393 King St E unitl, Toronto, ON M5A 1L3	Malcolm, Kelly	Clean store.	SPRING20

10. List all transactions carried out by "Toronto Convenience".

```
SELECT t.PKTrans_Id, sd.PName Product, td.Total_Cost, t.Trans_Date
FROM Transaction t JOIN Transaction_Details td ON t.PKTrans_Id=td.FKTrans_Id
JOIN Stock_Details sd ON td.FKStock_Id=sd.PKStock_Id
JOIN Store s ON t.FKS_Id=s.PKS_Id
WHERE s.PKS_Id=1002;
```

	PKTRANS_ID	PRODUCT	TOTAL_COST	TRANS_DATE
1	1001	Milk	99.5	04-APR-24
2	1002	Candy Bars	250	04-APR-24
3	1005	Bread	250	20-MAR-24

7 Conclusion

One of the primary challenges encountered during the project was the complexity and interconnectedness of various entities and their attributes. Ensuring that each entity was thoroughly planned, along with defining their attributes and establishing clear connections between them, was absolutely a complicated task. To overcome this challenge, we utilized techniques such as entity-relationship modeling to visually represent the connections between entities, ensuring clarity and coherence in the database structure. Hence, the efforts of comprehending every relationship enabled us to successfully navigate the complexities in database management and create a well-structured database.

In conclusion, this group project provided an opportunity for us to demonstrate and enhance the knowledge and skills we gained in the course -- Introduction to Database Concept. Through planning and creating, we aimed to develop a database that met the needs of convenience stores. The project facilitated the development of problem-solving skills in database management. Moreover, it served as a tool for reflection, allowing us to identify and address weaknesses while reinforcing our strengths. For instance, we distributed our work according to our strengths. The student who is good at brainstorming works for data requirements; The student who is good at logical thinking works for conceptual data modeling; Students who are good at typing code work for physical database design and query results. As we conclude this project, we recognize the importance of continuous improvement to meet the evolving demands of the database industry.

8 Reference

- Fake Canada Phone Numbers - Random Phone Number Generator - Fake Number. (n.d.). Fakenumber.org. Retrieved April 6, 2024, from https://fakenumber.org/canada
- Random Email Generator - Text Tools. (n.d.). Texttools.org. Retrieved April 6, 2024, from https://texttools.org/random-email-generator#google_vignette
- Canadian name generator. Canadian names. (n.d.). Retrieved April 6, 2024, from https://www.fantasynamegenerators.com/canadian-names.php.
- Textbook: Joan Casteel. 2016. Oracle 12c: SQL, 3rd Edition.
- https://static.thenounproject.com/png/252728-200.png
- https://png.pngtree.com/png-vector/20220706/ourmid/pngtree-project-management-png-image_5687733.png