INFS7901 Project Report

Name: Ruhan Pu
ID: s4677896
Email: r.pu@uqconnect.edu.au

UPDATE based on the proposal feedback:

1) Add FK in SQL dump.
2) In table `Customer`, change column name `VipType` to `VipID`.
3) Keep entity 'Product'. Table `OrderItem` shows the details of each order. We can see what items are included in an order. Table `Product` stores the information of all the products in stock, while some products are never sold, so they will be recorded in the `Product` table.

1. A brief statement on the aim and background of the project.

There are thousands of orders for a large shop in their daily running. Therefore, it is essential for them to import a database system to manage the messy data for their products and customers. With the help of this system, sellers can easily understand the sales and customer purchasing preferences for a specific period. Therefore, they can adjust their selling strategies in time and run their stores better.

This project creates a database system for shops. By using this database system, the managers are able to record the information about the sales and customers. This system will focus on product categories, sales, customer purchase history, VIP tiers, and more. In this model, users are able to add, delete or update information about products and customers. In addition, they are able to check the order information and turnover. Therefore they could better manage the shop.

2. Query Demonstration: (at least 5)

a. Join query

Join table `VIP` on table `Customer` using attribute `VipID`, so that the user can check what discount the customer has.

CustomerID ▲ 1	FirstName	LastName	VipType	Discounts
1	Luo	Valley	Gold	0.9
2	Jonathan	Tincob	Silver	0.95
3	Elena	Udrea	Bronze	1
4	Olivia	Abe	Platinum	0.8
5	Renata	Lucia	Gold	0.9
6	Martin	Wat	Diamond	0.85
7	Harry	Crofts	Silver	0.95
8	Jamie	Carllyle	Bronze	1
9	Mike	House	Bronze	1
10	Lydia	Daisy	Gold	0.9

Code:

```
SELECT c.CustomerID, c.FirstName, c.LastName, v.VipType, v.Discounts
FROM `Customer` c
Join VIP v USING (VipID)
ORDER BY c.CustomerID
```

b. Aggregation query (functions such as min, max, average or count)

Calculate the total amounts of all orders.

sales
98.80000114440918

Code:

```
SELECT SUM(Amounts) as sales
FROM `Orders`
WHERE 1
```

c. Aggregation with group-by (aggregated value for each group)

Calculate the total amounts of different order types.

OrderType	sales
Offline	82.80000114440918
Online	16

Code:

```
SELECT OrderType, sum(amounts) as sales
FROM `Orders`
GROUP BY OrderType
```

d. Delete operation with Cascade

When the user deletes an order in table `orders`, the relative tuple will be deleted automatically in table `Offline` or `Online`, and table `OrderItem`.

Initial tables and tables after deleting tuple 1:

Table `Orders`:

OrderID	Date	Amounts	OrderType	AddedPoints	CustomerID
1	2021-11-02	8	Offline	8	3
2	2021-12-28	11	Offline	11	1
3	2022-01-02	6	Online	6	9
4	2022-02-18	6	Offline	6	4
5	2022-03-16	31.2	Offline	31	10
6	2022-03-25	8	Online	8	2
7	2022-04-04	2	Online	2	3
8	2022-04-13	26.6	Offline	26	5

OrderID	Date	Amounts	OrderType	AddedPoints	CustomerID
2	2021-12-28	11	Offline	11	1
3	2022-01-02	6	Online	6	9
4	2022-02-18	6	Offline	6	4
5	2022-03-16	31.2	Offline	31	10
6	2022-03-25	8	Online	8	2
7	2022-04-04	2	Online	2	3
8	2022-04-13	26.6	Offline	26	5

Table `OrderItem`:

OrderID	ItemName	Quantity	Price	ProductID
1	Yoghurt	2	8	5
2	Shampoo	1	11	7
3	Carrot	2	4	1
3	Patato	1	2	4
4	Butter	1	6	6
5	Strauberry	2	31.2	3
6	Yoghurt	2	8	5
7	Tomato	2	27.6	2
8	Shampoo	1	11	7
8	Strawberry	1	15.6	3

OrderID	ItemName	Quantity	Price	ProductID
2	Shampoo	1	11	7
3	Carrot	2	4	1
3	Patato	1	2	4
4	Butter	1	6	6
5	Strauberry	2	31.2	3
6	Yoghurt	2	8	5
7	Tomato	2	27.6	2
8	Shampoo	1	11	7
8	Strawberry	1	15.6	3

Table `Offline`:

OrderID	PayMethod
1	Card
2	Cash
4	Card
5	Card
8	Cash

OrderID	PayMethod
2	Cash
4	Card
5	Card
8	Cash

Code:

```
DELETE FROM `Orders`
WHERE OrderID = 1
```

e. Update operation

Update the phone number of customer 1.

Initial table:

CustomerID	FirstName	LastName	Phone
1	Luo	Valley	(04) 9126 4896

After updating:

CustomerID	FirstName	LastName	Phone
1	Luo	Valley	(04) 1198 7456

Code:

```
SELECT CustomerID, FirstName, LastName, Phone
FROM `Customer`
WHERE CustomerID = 1

UPDATE `Customer`
SET Phone = "(04) 1198 7456"
WHERE CustomerID = 1
```

3. A conclusion that briefly describes the following:

- Roughly how much time did you spend on each part of the project?
  At the beginning of the project, I spent one day conceiving this project, such as the domain, structure, etc. Then, I spent most of my time designing the E/R diagram and mapping it. It took me about three or four days to finish it, and I also spend some time reviewing the lecture before doing this. Creating all tables and popularising them in SQL is much easier as long as you learn the lecture. It took me several hours to finish. Lastly, I spend around two days writing the queries and final report.

- What aspects do you like the most and the least in this project?
  My favourite part is designing and mapping the E/R diagram. I enjoy the process of brainstorming. It is time-consuming, but I was quite excited when I finished all these works. The aspect I like the least is popularizing the tables, especially the `Customer` table because it is hard and boring to write the information like name and address.

- What helped you learn the best in the project?
  The most helpful resources are lecture slides because they are very detailed and informative. I could find almost everything I need on them. In addition, I used to watch teaching videos by Mosh, and learned useful query language, which helps me to understand the lecture about SQL. I also search google when I met some problems with using SQL to create tables.

- What suggestions do you have for a beginner in database design?
  I think the beginner should be patient and careful when doing a database design. Sometimes relations between different entities are complex, and it takes time to think about it. Therefore, it is important to learn basic knowledge and understand their implications, because this can save time when designing a new database. When creating tables, they need to pay attention to the syntax, because if they miss punctuation, the PHP might show errors.