Project Name: **Digitech99+**

Lanshun Yuan
20411936

Module code: BUSI2118

Word Count: 2973

Content

1. Background

Notting’s Taste is a leading large-scale beverage company in Ningbo. Since its establishment in 2012, the company has focused on fruit teas and milk teas. Driven by product innovation, Notting’s Taste launched a new line of pre-made tea products in September 2022. This line formulated based on popular taste preferences in the market and can be made at home. Currently, the company expanded from 15 stores in Ningbo to 175 stores nationwide. Although the pre-made tea sells well, in recent years, the income of various regional stores has been unstable and the profit margin has not reached the expected levels. The app order entries in the database have not been updated, which has prevented the implementation of dynamic management, leading to poor overall operational efficiency. The current database is unable to accurately analyze the specific causes of unstable profits, thus preventing strategic adjustments to enhance market share. Currently, the company’s profitability is insufficient, and shareholder satisfaction is low. Therefore, the company necessitates a novel database to bolster profitability and market share from three vantage points: firstly, the database must conduct analyses on third-party entities to bolster company operations; secondly, it should efficaciously scrutinize the operational dynamics of all extant stores to aid headquarters in store management; lastly, database analytics can assess the operational performance of individual products, facilitating more judicious allocation of product resources by the company.

2. Conceptual database design

1) Business rule

- When products are made, the materials are automatically deducted from the warehouse.
- When a store receives supplies from a supplier, the warehouse automatically updates to reflect the increased materials.
- When a customer places an order and chooses delivery, the logistics provider automatically receives the customer's information.
- When a customer places an order and chooses to pick up, the customer automatically receives the pickup number.

- In the four channels, the offline delivery means customers book the product and get product by delivery; the online pickup means customers take orders by apps and get product in the shop.
- The head office will automatically handle product discounts, advertisement, and supplier rating. The advertising and discount system, supplier rating only reflects the data received by the store and does not represent the complete system.
- Product discounts require daily manual confirmation by employees.
- Invoices are automatically generated upon completion of material and customer orders, and invoice details are automatically recorded in the financial ledger.
- Each shop has only one manager. All other staff can check the warehouse of their shop, send the material order and handle customer order.
- The database updates in real-time and maintains historical data; however, the attached database only displays one day's data as a sample.

2) System definition

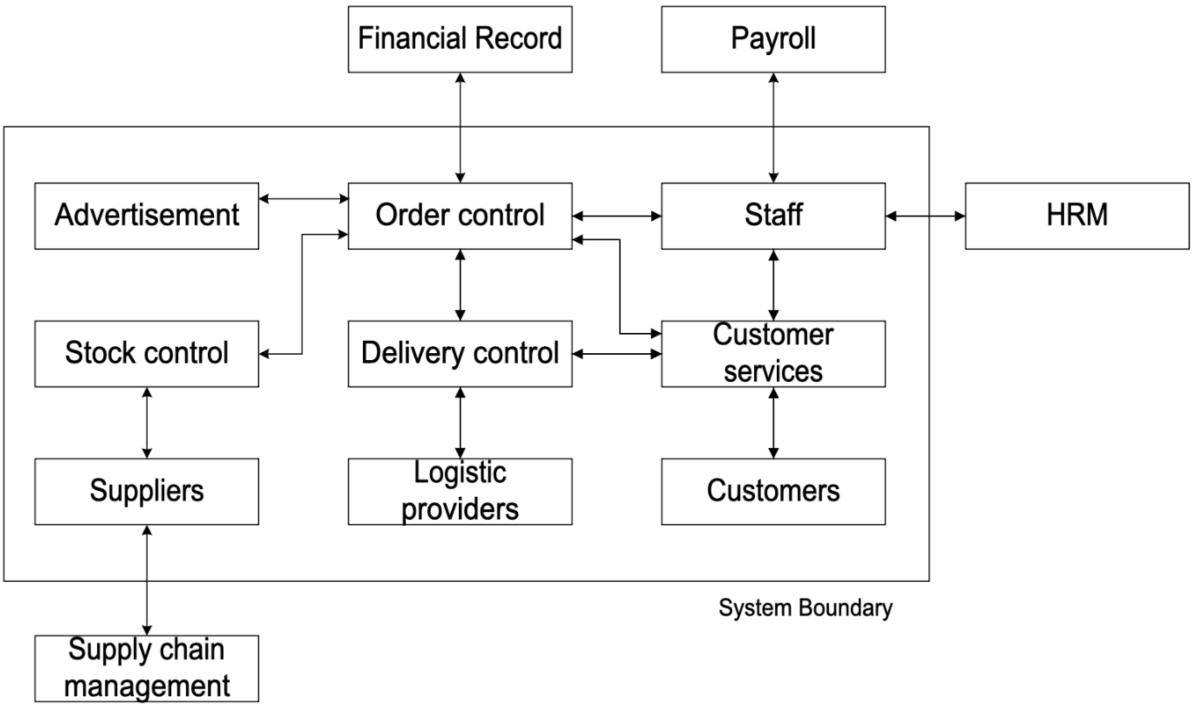


Fig2.1

As shown in Fig2.1, the system is a database of beverage shops operation system, which mainly handles six modules: order control, customer services, logistic control, stock control, advertisement.

There are four major user views, which are:

Customer

When a customer registers with Notting's Taste, the data stored includes the customer ID, name (first name and last name), contact number, gender, and age. Customers can track their orders, track order delivery information, and give feedback on their orders.

Staff

The data required on employees includes staff ID, name (first and last name), gender, position, contact number, hire date, end date, and the ID of shop where he/she works. Staff can manage orders, manage inventory, manage order delivery, and receive customer feedback.

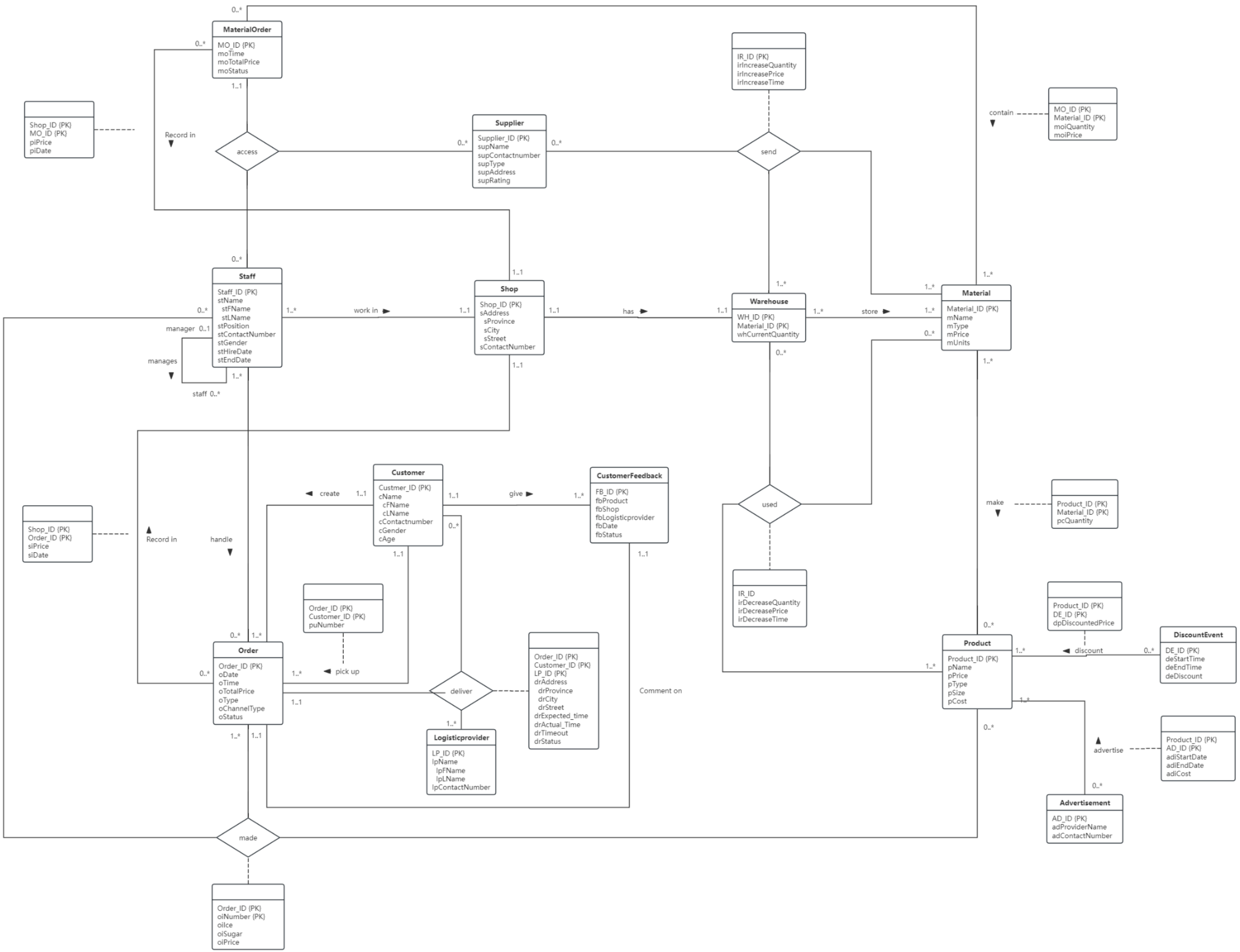
Supplier

The data required on suppliers includes supplier ID, company name, contact number of manager and address. Suppliers can track material orders.

Logistic provider

The data required on logistics provider includes logistics provider ID, company name, contact number. Logistics providers can track order delivery information.

3) ER diagram



0..*
MaterialOrder
MO_ID (PK)
moTime
moTotalPrice
moStatus
0..*
1..1
Shop_ID (PK)
MO_ID (PK)
piPrice
piDate
Record in
access
Supplier
Supplier_ID (PK)
supName
supContactnumber
supType
supAddress
supRating
0..*
0..*
IR_ID (PK)
irIncreaseQuantity
irIncreasePrice
irIncreaseTime
send
contain
MO_ID (PK)
Material_ID (PK)
moiQuantity
moiPrice
0..*
1..1
Staff
Staff_ID (PK)
stName
stFName
stLName
stPosition
stContactNumber
stGender
stHireDate
stEndDate
0..*
manager 0..1
manages
1..*
staff 0..*
1..*
work in
1..1
Shop
Shop_ID (PK)
sAddress
sProvince
sCity
sStreet
sContactNumber
1..1
1..1
has
1..1
Warehouse
WH_ID (PK)
Material_ID (PK)
whCurrentQuantity
1..*
0..*
1..*
store
1..*
1..*
0..*
1..*
Material
Material_ID (PK)
mName
mType
mPrice
mUnits
1..*
Customer
Custmer_ID (PK)
cName
cFName
cLName
cContactnumber
cGender
cAge
create
1..1
1..1
give
1..*
0..*
1..1
CustomerFeedback
FB_ID (PK)
fbProduct
fbShop
fbLogisticprovider
fbDate
fbStatus
1..1
used
make
Product_ID (PK)
Material_ID (PK)
pcQuantity
Shop_ID (PK)
Order_ID (PK)
siPrice
siDate
Record in
handle
IR_ID
irDecreaseQuantity
irDecreasePrice
irDecreaseTime
Order_ID (PK)
Customer_ID (PK)
puNumber
pick up
Product_ID (PK)
DE_ID (PK)
dpDiscountedPrice
0..*
0..*
1..*
Order
Order_ID (PK)
oDate
oTime
oTotalPrice
oType
oChannelType
oStatus
0..*
1..*
1..1
deliver
Order_ID (PK)
Customer_ID (PK)
LP_ID (PK)
drAddress
drProvince
drCity
drStreet
drExpected_time
drActual_Time
drTimeout
drStatus
Comment on
1..*
Product
Product_ID (PK)
pName
pPrice
pType
pSize
pCost
1..*
discount
0..*
DiscountEvent
DE_ID (PK)
deStartTime
deEndTime
deDiscount
1..*
1..*
1..1
Logisticprovider
LP_ID (PK)
lpName
lpFName
lpLName
lpContactNumber
0..*
1..*
advertise
Product_ID (PK)
AD_ID (PK)
adiStartDate
adiEndDate
adiCost
0..*
Advertisement
AD_ID (PK)
adProviderName
adContactNumber
made
Order_ID (PK)
oiNumber (PK)
oiIce
oiSugar
oiPrice

4) Data dictionary

Entity	Attribute	Description	Data Type & Length	Nulls	Multi-Valued
Customer	Customer_ID	Uniquely identifies a customer	10 variable characters	No	No
	cFName	First name of customer	50 variable characters	No	No
	cLName	Last name of customer	50 variable characters	No	No
	cContactNumber	Phone number of customer	15 variable characters	No	Yes
	cGender	Gender of customer. M = Male, F = Female, Null = Unknown.	1 character (M or F)	Yes	No
	cAge	Age of customer	integer	No	No
Staff	Staff_ID	Uniquely identifies a member of staff	10 variable characters	No	No
	Shop_ID	The shop where the staff work. Foreign key, associated with the shop table	10 variable characters	No	No
	stFName	First name of staff	50 variable characters	No	No
	stLName	Last name of staff	50 variable characters	No	No
	stGender	Gender of customer. M = Male, F = Female, Null = Unknown.	1 character (M or F)	Yes	No
	stPosition	Job title of member of staff	50 variable characters	No	Yes

	stContactNumber	Phone number of staff	15 variable characters	No	Yes
	stHire_Date	Date of hire of member of staff	Datetime	No	No
	stEnd_Date	Date of leave of member of staff. Null if staff still work	Datetime	Yes	No
Shop	Shop_ID	Uniquely identifies a shop	10 variable characters	No	No
	WH_ID	The warehouse of the shop. Foreign Key, associated with warehouse table	10 variable characters	No	No
	sProvince	State of shop	20 variable characters	No	No
	sCity	City of shop	20 variable characters	No	No
	sStreet	Detailed address of shop	60 variable characters	No	No
	sContactNumber	Phone number of shop	15 variable characters	No	Yes
Supplier	Supplier_ID	Uniquely identifies a supplier	10 variable characters	No	No
	supName	Name of supplier	60 variable characters	No	No
	supType	The goods or service provided by the supplier	20 variable characters	No	Yes
	supContactNumber	Phone number of supplier	15 variable characters	No	Yes
	supAddress	Company address of supplier	150 variable characters	No	Yes

	supRating	Employee ratings of suppliers from 1 to 6. 1 is the worst, 6 is the best.	integer	No	No
LogisticsProvider	LP_ID	Uniquely identifies a logistics provider	10 variable characters	No	No
	lpFName	First name of logistics provider	50 variable characters	No	No
	lpLName	Last name of logistics provider	50 variable characters	No	No
	lpContactNumber	Phone number of logistics provider	15 variable characters	No	Yes
Product	Product_ID	Uniquely identifies a product	10 variable characters	No	No
	pName	Name of product	50 variable characters	No	No
	pPrice	Unit sale price of product	decimal	No	No
	pType	Wheter the product is prepared tea. P = Yes, F = No	1 character (P or F)	No	No
	pSize	The size of the product. E.g. small, middle, large. Null if the product does not have size.	10 variable characters	Yes	No
	pCost	Unit cost of product. Sum of the price of	decimal	No	No

		material in one product.			
Advertisement	AD_ID	Uniquely identifies a promotion	10 variable characters	No	No
	adProviderName	Name or company name of the promotion provider	100 variable characters	No	No
	adContactNumber	Contact number of the promotion provider	15 variable characters	No	Yes
DiscountEvent	DE_ID	Uniquely identifies a discount event	10 variable characters	No	No
	deStartTime	The start time of the discount	datetime	No	No
	deEndTime	The end time of the discount	datetime	No	No
	deDiscount	The discount of the product in the event. E.g. 0.5=50% off, 0.7=30%0ff, 1= no discount	decimal	No	No
Material	Material_ID	Uniquely identifies a material	10 variable characters	No	No
	mName	Name of material	50 variable characters	No	No
	mType	Material type, e.g. food material, packing material	50 variable characters	No	No
	mPrice	Unit price of material	decimal	No	No
	mUnits	The measuring unit of material	10 variable characters	No	No

Order	Order_ID	Uniquely identifies a order	10 variable characters	No	No
	Customer_ID	The customer of the order. Foreign key, associated with the product table	10 variable characters	No	No
	Staff_ID	The staff of the order. Foreign key, associated with the product table	10 variable characters	No	No
	oDate	The date when the order takes place	datetime	No	No
	oTime	The time when the order takes place	datetime	No	No
	oTotalPrice	total after discount sale price of an whole order. The sum of product after discount price in one order.	decimal	No	No
	oType	The way to place an order. On = Online, Off = offline	10 variable characters	No	No
	oChannelType	The way to delivery an order. P = pick up by themself, D = delivery	1 character (P or D)	No	No

MaterialOrder	MO_ID	Uniquely identifies a material order	10 variable characters	No	No
	Supplier_ID	The supplier who provide the material. Foreign key, associated with the supplier table	10 variable characters	No	No
	Staff_ID	The staff who order the material. Foreign key, associated with the staff table	10 variable characters	No	No
	moTime	The time when the material order takes place	datetime	No	No
	moTotalPrice	Total price of the material	decimal	No	No
	moStatus	The status of the material order. P = Pending D = Done	1 character (P or D)	No	No
CustomerFeedback	FB_ID	Uniquely identifies a customer feedback	10 variable characters	No	No
	Customer_ID	The customer who give the feedback. Foreign key, associated with the customer table	10 variable characters	No	No
	Order_ID	Customer feedback	10 variable characters	No	No

		problems with orders. Foreign key, associated with the order table			
	fbProduct	Customer feedback of product. The rating is an integer from 1 to 6, with 1 being very dissatisfied and 6 being very satisfied	Integer	No	No
	fbShop	Customer feedback of shop. The rating is an integer from 1 to 6, with 1 being very dissatisfied and 6 being very satisfied	Integer	No	No
	fbLogisticProvider	Customer feedback of logistic provider. The rating is an integer from 1 to 6, with 1 being very dissatisfied and 6 being very satisfied	Integer	No	No
	fbDate	Date of submission of the feedback	datetime	No	No

	fbStatus	The processing status of the feedback. P = Pending S = Solve	1 character (P or S)	No	No
Warehouse	WH_ID	Uniquely identifies a warehouse. Primary key	10 variable characters	No	No
	Material_ID	The id of material. Primary key and foreign key, associated with the material table	decimal	No	No
	whCurrentQuantity	The current quantity of the material in the warehouse	decimal	No	No

Relationship	Attribute	Description	Data Type & Length	Nulls	Multi-Valued
AdvertisementItem	Product_ID	Primary key and foreign key, associated with the product table	10 variable characters	No	No
	AD_ID	Primary key and foreign key, associated with the promotion table	10 variable characters	No	No
	adiStartDate	Start time of the promotion	datetime	No	No
	adiEndDate	End time of the promotion. Null if	datetime	Yes	No

		the promotion is continuing			
	adiCost	The promotion cost of each product	decimal	No	No
DiscountedItem	DE_ID	Primary key and foreign key, associated with the DiscountEvent table	10 variable characters	No	No
	Product_ID	Primary key and foreign key, associated with the Product table	10 variable characters	No	No
	dpDiscountedPrice	The after discount price of the product in the discount event			
ProductComponent	Product_ID	Primary key and foreign key, associated with the material table	10 variable characters	No	No
	Material_ID	Primary key and foreign key, associated with the material table	10 variable characters	No	No
	pcQuantity	the quantity of a particular material in one product	decimal	No	No
OrderItem	Order_ID	Primary key and foreign key, associated with the orders table	10 variable characters	No	No

	oiNumber	Primary key, uniquely identified a product in an order. The drinks number in an order.	integer	No	No
	Product_ID	Foreign key, associated with the product table	10 variable characters	No	No
	Staff_ID	The id of staff who make the drink. Foreign key, associated with staff table.	10 variable characters	No	No
	oiIce	The degree of ice of the product. 10=normal ice, 5=less ice, 0=no ice	Integer	No	No
	oiSugar	The degree of sugar of the product. 10=normal sugar, 7=less sugar, 5=half sugar, 3=low sugar, 0=no sugar	integer	No	No
	oiPrice	The total after discount price of a product in the order.	decimal	No	No
MaterialOrderItem	MO_ID	Primary key and foreign key, associated with	10 variable characters	No	No

		the MaterialOrder table			
	Material_ID	Primary key and foreign key, associated with the material table	10 variable characters	No	No
	moiQuantity	Sale volume of a particular matreial in a particular order	integer	No	No
	moiPrice	total price of material in one order	decimal	No	No
PickUpRecord	Order_ID	Primary key and foreign key, associated with the order table	10 variable characters	No	No
	Customer_ID	Primary key and foreign key, associated with the customer table	10 variable characters	No	No
	puNumber	The pick up number of the order	10 variable characters	No	No
DeliveryRecord	LP_ID	Primary key and foreign key, associated with the LogisticProvider table	10 variable characters	No	No
	Order_ID	Primary key and foreign key,	10 variable characters	No	No

		associated with the order table			
	Customer_ID	Primary key and foreign key, associated with the customer table	10 variable characters	No	No
	drprovince	State of delivery address	20 variable characters	No	No
	drCity	City of delivery address	20 variable characters	No	No
	drStreet	Detailed delivery address	60 variable characters	No	No
	drExpectedTime	Expected time the supplier order delivered	datetime	No	No
	drActualTime	Actual time the order delivered	datetime	No	No
	drTimeout	Y = timeout, N = not timeout.	1 character (Y or N)	No	No
	drStatus	current status of the task. (C = cancelled, D = delivered)	1 character (C or D)	No	No
InventoryInRecord	IR_ID	Uniquely identified a inventory record	10 variable characters	No	No
	Material_ID	Foreign key, associated with the material table	10 variable characters	No	No
	WH_ID	Foreign key, associated with the warehouse table	10 variable characters	No	No

	Supplier_ID	Foreign key, associated with the supplier table.	10 variable characters	Yes	No
	IrIncreaseQuantity	the quantity of material increased	decimal	No	No
	irIncreasePrice	The total price of material increase	decimal	No	No
	irIncreaseTime	The time when the material increase	datetime	No	No
InventoryOutRecord	IR_ID	Uniquely identified a inventory record	10 variable characters	No	No
	Material_ID	Foreign key, associated with the material table	10 variable characters	No	No
	WH_ID	Foreign key, associated with the warehouse table	10 variable characters	No	No
	Product_ID	Foreign key, associated with the product table.	10 variable characters	Yes	No
	IrDecreaseQuantity	the quantity of material decreased	decimal	No	No
	irDecreasePrice	The total price of material decrease	decimal	No	No
	irDecreaseTime	The time when the material decrease	datetime	No	No
PurchaseInvoice	Shop_ID	Primary key and foreign key,	10 variable characters	No	No

		associated with the shop table			
	MO_ID	Primary key and foreign key, associated with the MaterialOrder table	10 variable characters	No	No
	piPrice	Purchase price of the material order	decimal	No	No
	piDate	invoicing date of the material order	datetime	No	No
SaleInvoice	Shop_ID	Primary key and foreign key, associated with the shop table	10 variable characters	No	No
	Order_ID	Primary key and foreign key, associated with the order table	10 variable characters	No	No
	siPrice	sale price of the order	decimal	No	No
	siDate	invoicing date of the order	datetime	No	No

3. Logical database design

1) Data schema

Customer (Customer_ID, cFName, cLName, cContactNumber, cGender, cAge) **Primary Key** Customer_ID **Alternate Key** cContactNumber	**Staff** (Staff_ID, Shop_ID, stFName, stLName, stGender, stPosition, stContactNumber, stHireDate, stEndDate) **Primary Key** Staff_ID

	Foreign Key Shop_ID **references** Shop(Shop_ID) **Alternate Key** stContactNumber
Order (Order_ID, Customer_ID, Staff_ID, oDate, oTime, oTotalPrice, oType, oChannelType) **Primary Key** Order_ID **Foreign Key** Customer_ID **references** Customer(Customer_ID) **Foreign Key** Staff_ID **references** Staff(Staff_ID) **Alternate Key** Customer_ID, oTime **Alternate Key** Staff_ID, oTime	**OrderItem** (Order_ID, oiNumber, Product_ID, Staff_ID, oiIce, oiSugar, oiPrice) **Primary Key** Order_ID, oiNumber **Foreign Key** Order_ID **references** Order(Order_ID) **Foreign Key** Product_ID **references** Product(Product_ID) **Foreign Key** Staff_ID **references** Staff(Staff_ID)
Shop (Shop_ID, WH_ID, sProvince, sCity, sStreet, sContactNumber) **Primary Key** Shop_ID **Foreign Key** WH_ID **references** WH(WH_ID) **Alternate Key** sContactNumber	**Supplier** (Supplier_ID, supName, supType, supContactNumber, supAddress, supRating) **Primary Key** Supplier_ID **Alternate Key** supContactNumber
LogisticsProvider (LP_ID, lpFName, lpLName, lpContactNumber) **Primary Key** LP_ID **Alternate Key** lpContactNumber	**Product** (Product_ID, pName, pPrice, pType, pSize, pCost) **Primary Key** Product_ID **Alternate Key** pName, pSize, pPreType
Adevertisement (AD_ID, adProviderName, adContactNumber) **Primary Key** AD_ID **Alternate Key** adContactNumber	**AdvertisementItem** (Product_ID, AD_ID, adiStartDate, adiEndDate, adiCost) **Primary Key** Product_ID, AD_ID **Derived** duration (adiEndDate-adiStartDate)
DiscountEvent (DE_ID, deStartTime, deEndTime, deDiscount) **Primary Key** DE_ID **Derived** duration (deEndTime-deStartTime)	**DiscountedItem** (Product_ID, DE_ID, dpDiscountedPrice) **Primary Key** Product_ID, DE_ID

Material (Material_ID, mName, mType, mPrice, mUnits) **Primary Key** Material_ID **Alternate Key** mName, mPrice	**ProductComponent** (Product_ID, Material_ID, pcQuantity) **Primary Key** Product_ID, Material_ID **Foreign Key** Product_ID **references** Product(Product_ID) **Foreign Key** Material_ID **references** Material(Material_ID)
MaterialOrder (MO_ID, Supplier_ID, Staff_ID, moTime, moTotalPrice, moStatus) **Primary Key** MO_ID **Foreign Key** Supplier_ID **references** Supplier(Supplier_ID) **Foreign Key** Staff_ID **references** Staff(Staff_ID) **Alternate Key** Supplier_ID, Staff_ID, moTime	**MaterialOrderItem** (MO_ID, Material_ID, moiQuantity, moiPrice) **Primary Key** MO_ID, Material_ID **Foreign Key** MO_ID **references** MO(MO_ID) **Foreign Key** Material_ID **references** Material(Material_ID)
SaleInvoice (Shop_ID, Order_ID, siPrice, siDate) **Primary Key** Shop_ID, Order_ID **Foreign Key** Shop_ID **references** Shop(Shop _ID)**Foreign Key** Order_ID **references** Order(Order_ID)	**PurchaseInvoice** (Shop_ID, MO_ID, piPrice, piDate) **Primary Key** Shop_ID, MO_ID **Foreign Key** Shop_ID **references** Shop(Shop _ID)**Foreign Key** MO_ID **references** MO(MO_ID)
CustomerFB (FB_ID, Customer_ID, Order_ID, fbProduct, fbShop, fbLogisticprovider, fbDate, fbStatus) **Primary Key** FB_ID **Foreign Key** Customer_ID **references** Customer(Customer_ID) **Foreign Key** Order_ID **references** Order(Order_ID) **Alternate Key** Customer_ID, Order_ID	**Warehouse** (WH_ID, Material_ID, whCurrentQuantity) **Primary Key** WH_ID, Material_ID **Foreign Key** Material_ID **references** Material(Material_ID)
InventoryRecord (IR_ID, Material_ID, WH_ID, Supplier_ID, Product_ID,	**DeliveryRecord** (LP_ID, Order_ID, Customer_ID, drprovince, drCity, drStreet,

irChangeType, irChangeQuantity, irPrice, irTime) **Primary Key** IR_ID **Foreign Key** Material_ID **references** Material(Material_ID) **Foreign Key** WH_ID **references** WH(WH_ID) **Foreign Key** Supplier_ID **references** Supplier(Supplier_ID) **Foreign Key** Product_ID **references** Product(Product_ID) **Alternate Key** Material_ID, WH_ID, irTime, irChangeType	drExpectedTime, drActualTime, drTimeout, drStatus) **Primary Key** LP_ID, Order_ID, Customer_ID **Foreign Key** LP_ID **references** LP(LP_ID) **Foreign Key** Order_ID **references** Order(Order_ID) **Foreign Key** Customer_ID **references** Customer(Customer_ID) **Derived** duration (DeliveryRecord.drActualTime-DeliveryRecord.drExpectedTime)

4. SQL

The SQL query is designed for customers to query their basic order information including their order items.

```
SELECT [Order].Order_ID,
    Product.pName AS ItemType,
    OrderItem.oiQuantity AS Quantity,
    DeliveryRecord.drStatus AS Status
FROM
    [Order], OrderItem, Product, DeliveryRecord
WHERE
    [Order].Order_ID = OrderItem.Order_ID
    AND [OrderItem].Product_ID = Product.Product_ID
    AND [Order].Order_ID = DeliveryRecord.Order_ID
    AND [Order].Customer_ID = [Enter your customer ID:];
```

Fig4.1

This figure illustrates the result of executing the SQL query with the customer ID "C001" as input.

Order_ID	ItemType	Quantity	Status
Or001	Silver Needle	1	D
Or001	Earl Grey Lavender	1	D
Or001	Earl Grey Lavender	2	D

Fig4.2

This SQL is created to fetch the discounted price of the certain product as if the current date.

```
SELECT
    Product.Product_ID,
    Product.pName,
    IIF(
        EXISTS(
            SELECT 1
            FROM DiscountEvent
            INNER JOIN DiscountedItem ON DiscountEvent.DE_ID = DiscountedItem.DE_ID
            WHERE DiscountEvent.deStartTime <= Date() AND DiscountEvent.deEndTime >= Date()
            AND DiscountedItem.Product_ID = Product.Product_ID
        ),
        (SELECT DiscountedItem.dpDiscountedPrice
         FROM DiscountedItem
         WHERE DiscountedItem.Product_ID = Product.Product_ID),
        Product.pPrice
    ) AS Price
FROM Product
WHERE Product.Product_ID = [Enter the Product_ID];
```

Fig4.3

This graph represents the output from running the SQL query with the product ID "P001" as input.

Product_ID	pName	Price
P001	Notting Tea	20

Fig4.4

This SQL query is designed to recommend the top five welcomed product to customers.

```
SELECT TOP 5
    Product.pType,
    Product.pName
FROM
    OrderItem
INNER JOIN
    Product ON [OrderItem].Product_ID = Product.Product_ID
WHERE
    Product.pType IN ('F', 'P')
GROUP BY
    Product.pType, Product.pName
ORDER BY
    SUM(OrderItem.oiQuantity) DESC;
```

Fig4.5

The image showcases the results identifying the top five most popular products among customers.

pType	pName
P	Earl Grey Lavender
P	Rose Petal Black
P	Notting Tea
P	Matcha Blast
P	Notting Jasmine Pearl

Fig4.6

This SQL query recommends the top three products that are most frequently purchased in combination with a specific item.

```sql
SELECT TOP 3
    oi.Product_ID AS RecommendedProductID,
    p.pName AS ProductName
FROM
    (OrderItem AS oi
INNER JOIN
    Product AS p ON oi.Product_ID = p.Product_ID)
INNER JOIN
    (SELECT Order_ID FROM OrderItem WHERE Product_ID = [SelectedProductID]) AS selectedOrders
ON
    oi.Order_ID = SelectedOrders.Order_ID
WHERE
    oi.Product_ID <> [SelectedProductID]
GROUP BY
    oi.Product_ID, p.pName
ORDER BY COUNT(*) DESC;
```

Fig4.7

The SQL query is executed with the input product ID "P001".

RecommendedProductID	ProductName
P121	Notting Tea
P134	Matcha Blast
P138	Rose Petal Black
P127	Oolong Supreme

Fig4.8

The SQL query is crafted for employees to retrieve statistics about orders they have handled.

```sql
SELECT
    [Order].Staff_ID,
    [Staff].Shop_ID,
    COUNT([Order].Order_ID) AS OrderCount,
    SUM([Order].oTotalPrice) AS TotalAmount
FROM
    [Order]
INNER JOIN
    Staff ON [Order].Staff_ID = Staff.Staff_ID
WHERE
    [Order].Staff_ID = [Enter your Staff_ID:]
GROUP BY
    [Order].Staff_ID, Staff.Shop_ID;
```

Fig4.9

The SQL query was run with the input "ST001" to retrieve specific information about the orders processed by the employee with Staff ID "ST001".

Staff_ID	Shop_ID	OrderCount	TotalAmount
ST001	S001	52	4219

Fig4.10

The SQL query is set up to enable employees to gather feedback on orders they have facilitated.

```
SELECT
    Order.Staff_ID,
    Order.Order_ID,
    NZ(CustomerFB.fbProduct, 'No Feedback') AS ProductFeedback,
    NZ(CustomerFB.fbShop, 'No Feedback') AS ShopFeedback
FROM
    [Order]
LEFT JOIN
    CustomerFB ON Order.Order_ID = CustomerFB.Order_ID
WHERE
    Order.Staff_ID = [Enter your Staff_ID:];
```

Fig4.11

The example output displayed in Graph 3-12, queried for Staff ID "ST001".

Staff_ID	Order_ID	ProductFee	ShopFeedba
ST001	Or1053	No Feedback	No Feedback
ST001	Or1101	No Feedback	No Feedback
ST001	Or1133	No Feedback	No Feedback
ST001	Or1188	4	6
ST001	Or1192	No Feedback	No Feedback
ST001	Or1194	5	4
ST001	Or1236	No Feedback	No Feedback
ST001	Or1240	No Feedback	No Feedback
ST001	Or1351	No Feedback	No Feedback
ST001	Or1360	No Feedback	No Feedback
ST001	Or1368	No Feedback	No Feedback
ST001	Or1392	No Feedback	No Feedback
ST001	Or1398	No Feedback	No Feedback
ST001	Or1442	No Feedback	No Feedback
ST001	Or1579	1	6

Fig4.12

The SQL query is designed for logistics providers to track performance by calculating the timeout rate of deliveries.

```
SELECT
    LP_ID,
    COUNT(*) AS TotalOrders,
    SUM(IIF(drTimeout = 'Y', 1, 0)) AS TimeoutOrders,
    FORMAT((SUM(IIF(drTimeout = 'Y', 1, 0)) / COUNT(*)) * 100, "0.00") & "%" AS TimeoutRate
FROM
    DeliveryRecord
WHERE
    LP_ID = [Enter your LP_ID:]
GROUP BY LP_ID;
```

Fig4.13

The Graph shows the result of the logistics provider ID "LP001".

LP_ID	TotalOrder:	TimeoutOrders	TimeoutRate
LP001	14	8	57.14%

Fig4.14

This SQL is structured to query the profit margin of the company's various products.

```
SELECT
    Product.Product_ID,
    Product.pName AS Product_Name,
    Product.pPrice AS Sales_Price,
    SUM(Materials.mPrice * ProductComponent.pcQuantity) AS Total_Cost,
    FORMAT((Product.pPrice - SUM(Materials.mPrice * ProductComponent.pcQuantity)) / Product.pPrice, "Percent") AS Profit_Margin_Percentage
FROM
    (Product INNER JOIN ProductComponent ON Product.Product_ID = ProductComponent.Product_ID)
INNER JOIN
    Materials ON ProductComponent.Material_ID = Materials.Material_ID
GROUP BY
    Product.Product_ID, Product.pName, Product.pPrice
ORDER BY
    (Product.pPrice - SUM(Materials.mPrice * ProductComponent.pcQuantity)) / Product.pPrice DESC;
```

Fig4.15

The accompanying output snapshot shows various products ranking their profit margins from high to low.

Product_ID	Product_Name	Sales_Price	Total_Cost	Profit
P010	Earl Grey Lavender	19	4.947	73.96%
P032	Mint Green	25	7.511	69.96%
P025	Chai Spice	19	5.849	69.22%
P050	Vanilla Rooibos	24	8.075	66.35%
P080	Bamboo Forest	25	8.476	66.10%
P013	Matcha Blast	20	6.824	65.88%
P118	Cranberry Splash	17	5.963	64.92%
P113	Licorice Leaf	24	8.452	64.78%
P049	Vanilla Rooibos	20	7.104	64.48%
P079	Bamboo Forest	18	6.407	64.41%
P027	Chai Spice	27	9.615	64.39%
P094	Pistachio Matcha	18	6.44	64.22%
P096	Pistachio Matcha	29	10.569	63.56%
P011	Earl Grey Lavender	25	9.235	63.06%
P112	Licorice Leaf	17	6.51	61.71%
P026	Chai Spice	21	8.115	61.36%
P040	Passion Fruit Oolong	20	7.879	60.61%
P104	Toffee Crunch	23	9.091	60.47%
P004	Jasmine Pearl	16	6.338	60.39%
P081	Bamboo Forest	29	11.539	60.21%

Fig4.16

The SQL is designed to calculate total sale amounts of the certain shop.

```
SELECT
    Shop_ID,
    SUM(siPrice) AS TotalSales
FROM
    SalesInvoice
WHERE
    Shop_ID = [Enter Shop_ID]
GROUP BY Shop_ID;
```

Fig4.17

The result for shop ID "S001" is shown in the accompanying output, indicating that the total sales amount to 4219.

Shop_ID	TotalSales
S001	4219

Fig4.18

This SQL is used to count the ordering channel and delivery channel for all orders of the company.

```
SELECT
    Order.oType,
    Order.oChannelType,
    COUNT(*) AS Order_Count
FROM
    [Order]
GROUP BY
    Order.oType, Order.oChannelType
ORDER BY
    Order.oType, Order.oChannelType;
```

Fig4.19

The graph displays the count of orders segmented by their ordering and channel types.

oType	oChannelType	Order_Count
Off	D	505
Off	P	1333
On	D	1085
On	P	1277

Fig4.20

This SQL query retrieves the current stock quantity for a specified Shop_ID and Material_ID.

```
SELECT
    WareHouse.whCurrentQuantity
FROM
    WareHouse
INNER JOIN
    Shop ON WareHouse.WH_ID = Shop.WH_ID
WHERE
    Shop.Shop_ID = [Enter Shop_ID] AND WareHouse.Material_ID = [Enter Material_ID];
```

Fig4.21

For the input Shop_ID of S001 and Material_ID of M001, this SQL query fetches the current stock quantity by joining the related shop and inventory records based on their warehouse association.

whCurrentQuantity
2211

Fig4.22

5. Data visualization

1) Use to decide product strategy:

This stacked bar chart illustrates the sales of pre-made and fresh tea in different provinces, and the corresponding cost at the same time, showing the profit and revenue gained in different provinces.

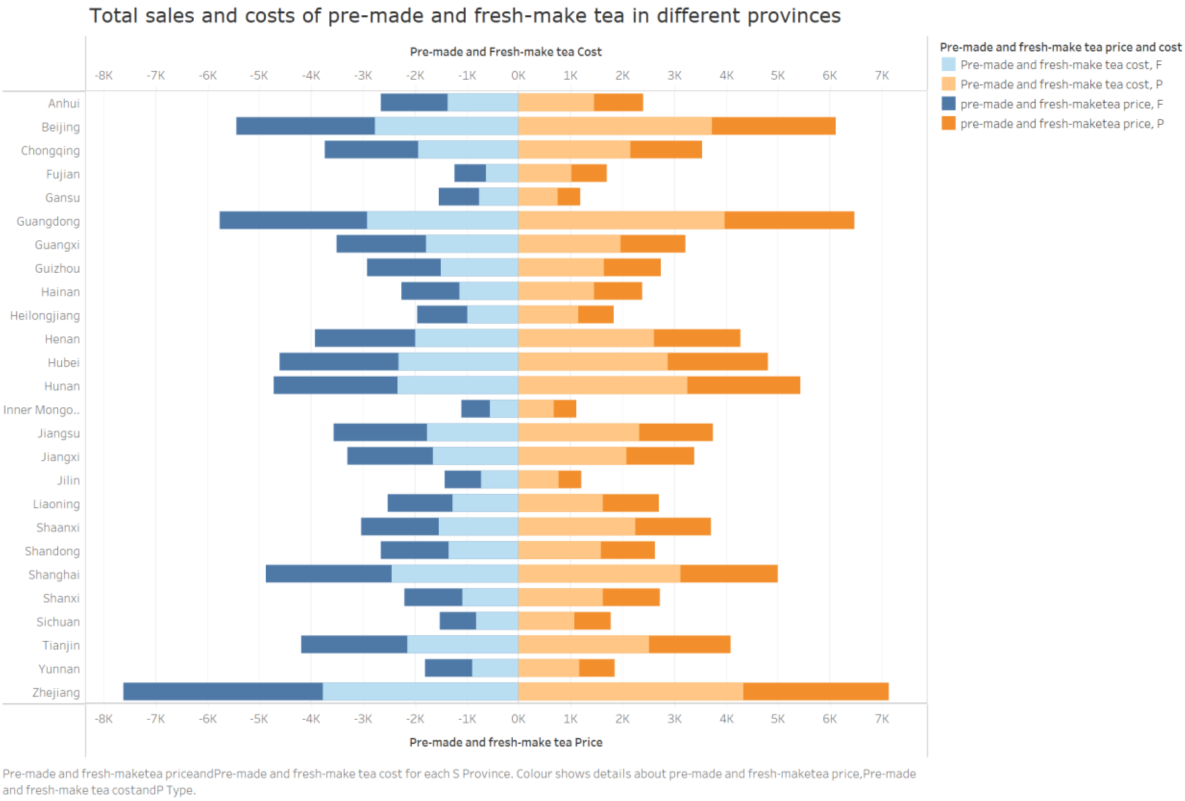


Fig5.1

This radar chart shows the gross profit margin of pre-made and fresh tea, the proportion of total profit, total revenue and the preference rate of customers, presenting the two type tea sales performance.

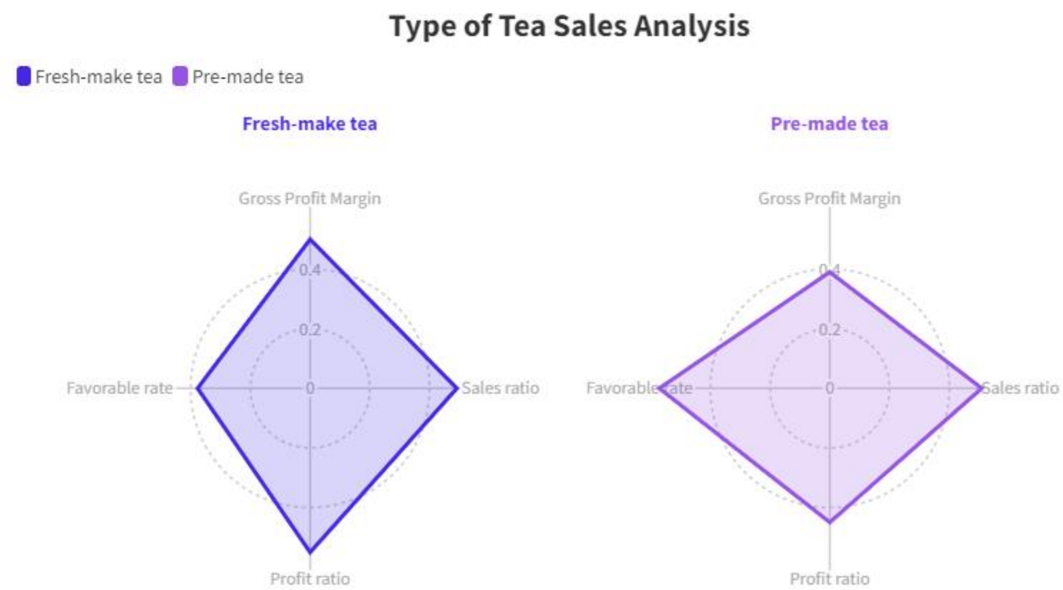


Fig5.2

This combined chart expresses sales volume and sales growth rate of pre-made [1]and fresh tea in 4 months*, showing the trend of customer choices.

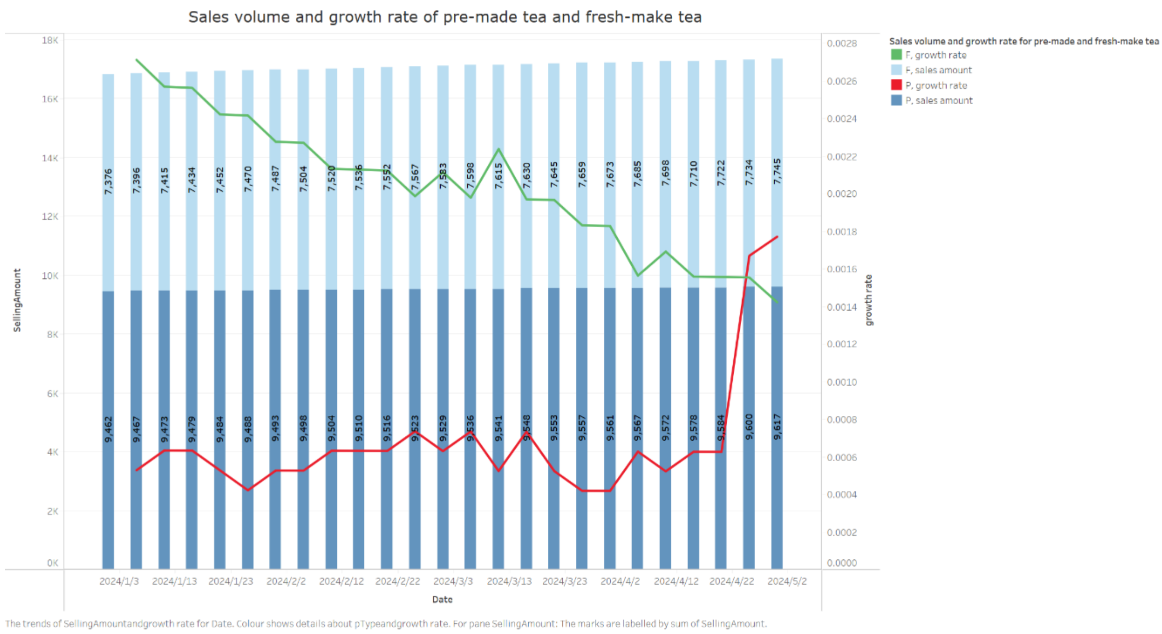


Fig5.3

This map distribution chart shows the customers’ preference for pre-made and fresh tea in each province for one-month*.

*: Our database only contain the sales data in one day, but to insure the accuracy of the analysis, we incorporate historical data from the previous months.

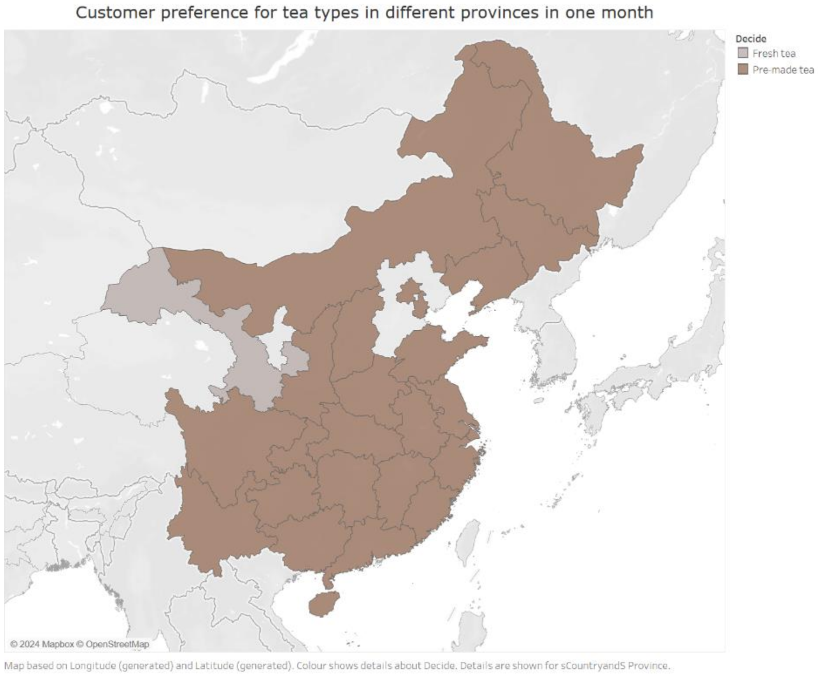


Fig5.4

This word cloud shows customer preferences for their products.

Fig5.5

2) Use to monitor shops:

The bar chart shows the sales of stores in different provinces compared to their current inventory, representing the shops' efficiency.

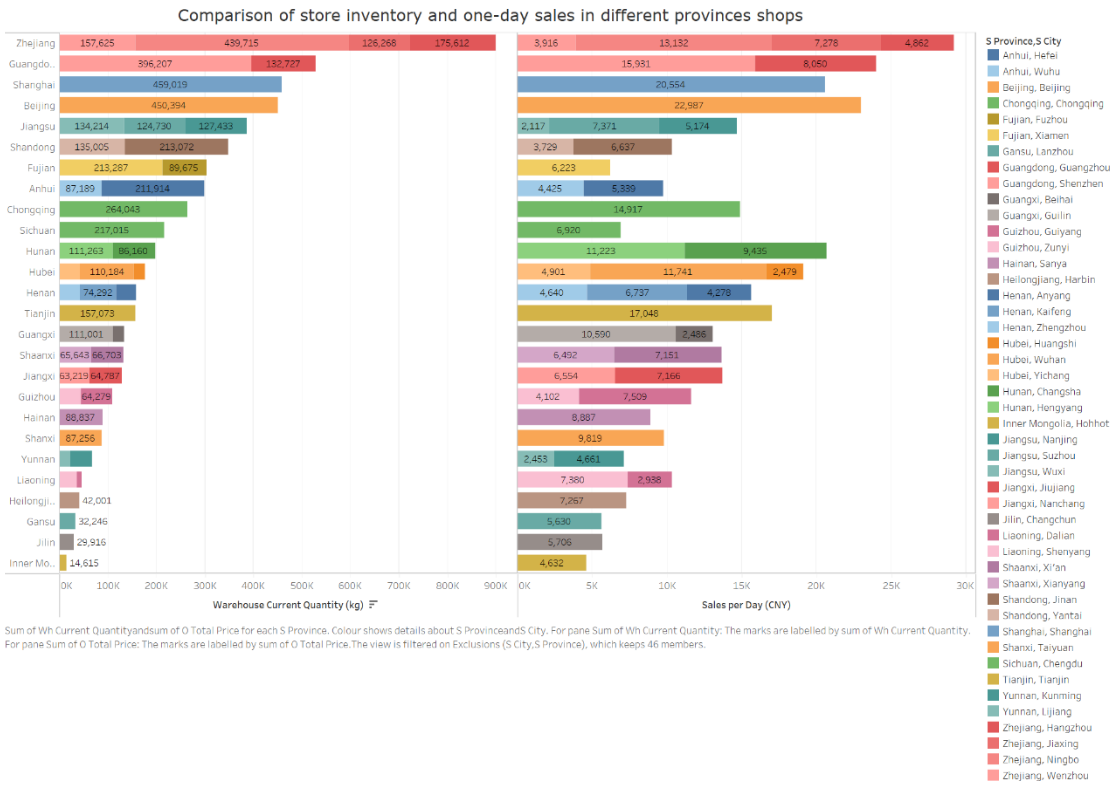


Fig5.6

This map shows the average ratings and distributions of stores in different provinces, presenting the shops' performance.

Fig5.7

This map shows the daily order volume distribution of stores in different provinces.

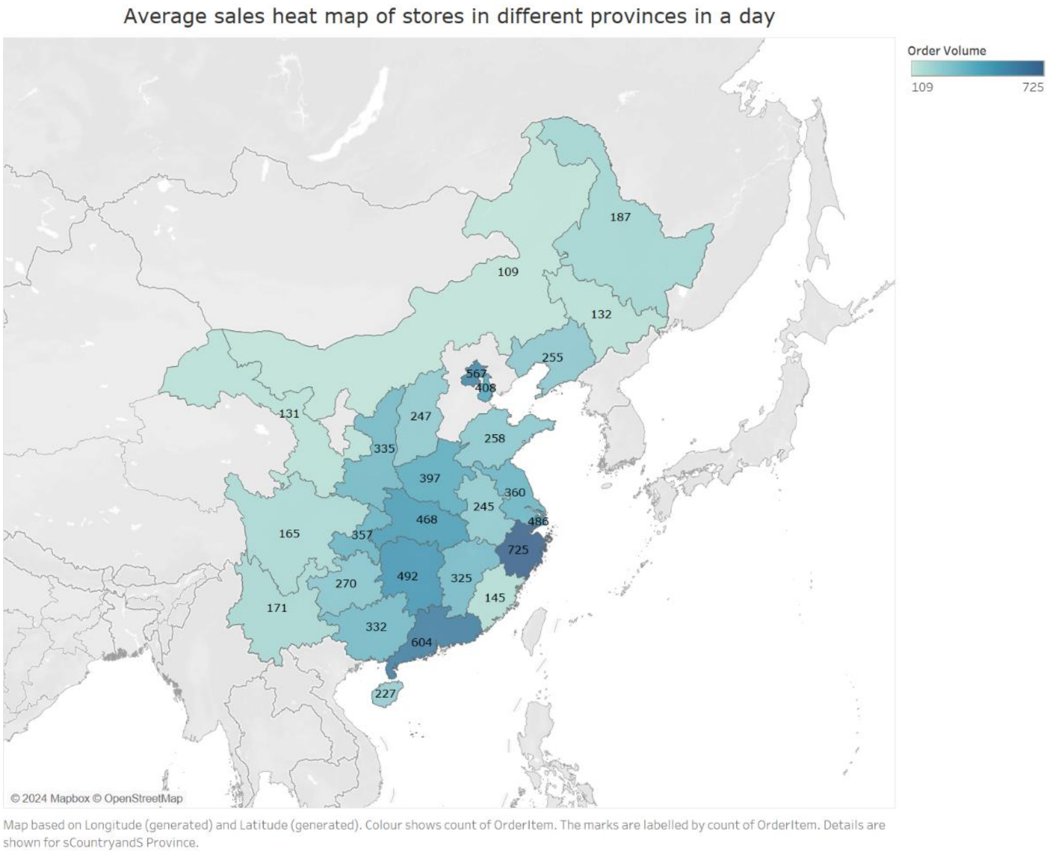


Fig5.8

3) Use to monitor different receive channels for products:

This heat table shows the ratings of products purchased from different channels at different times of the day, presenting customer satisfaction on different channels.

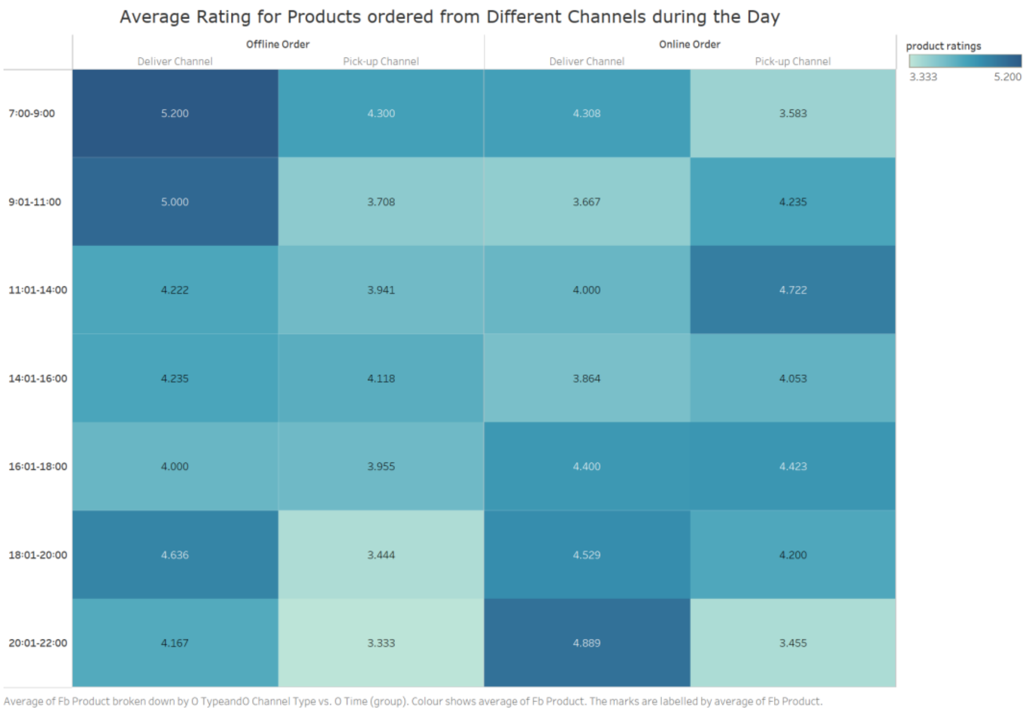


Fig5.9

This sankey illustrates the fluidity of channel selection in a month*, help to analyze customer preference options for channel.

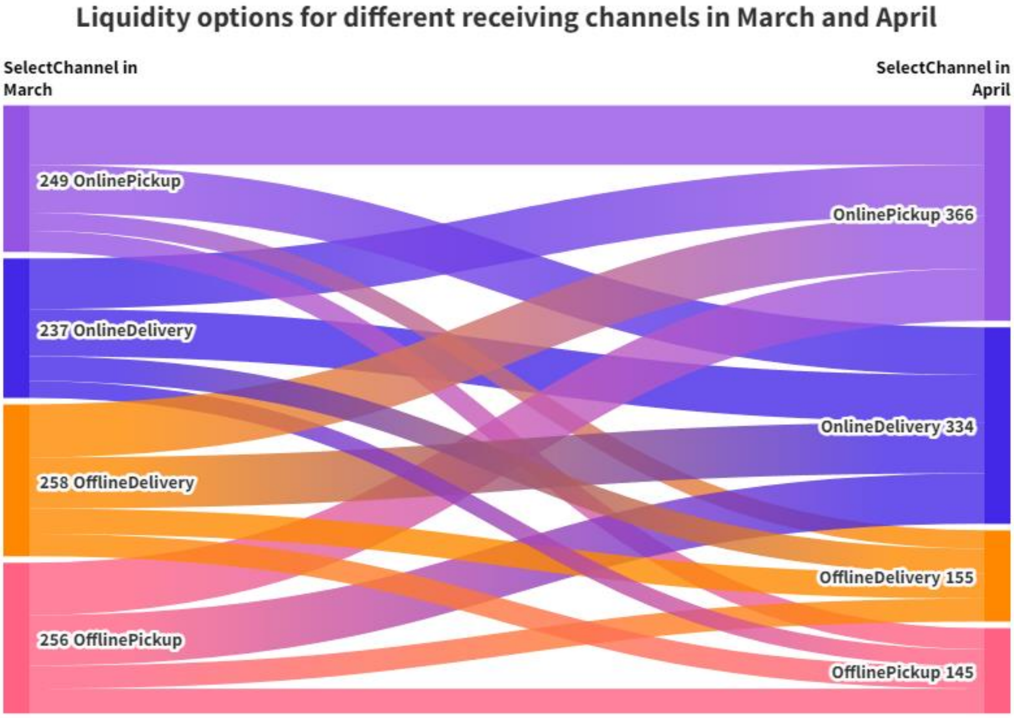


Fig5.10

This line chart shows the sales of each channel at different times of the day, showing the customer's preference for the channel at different times of the day.

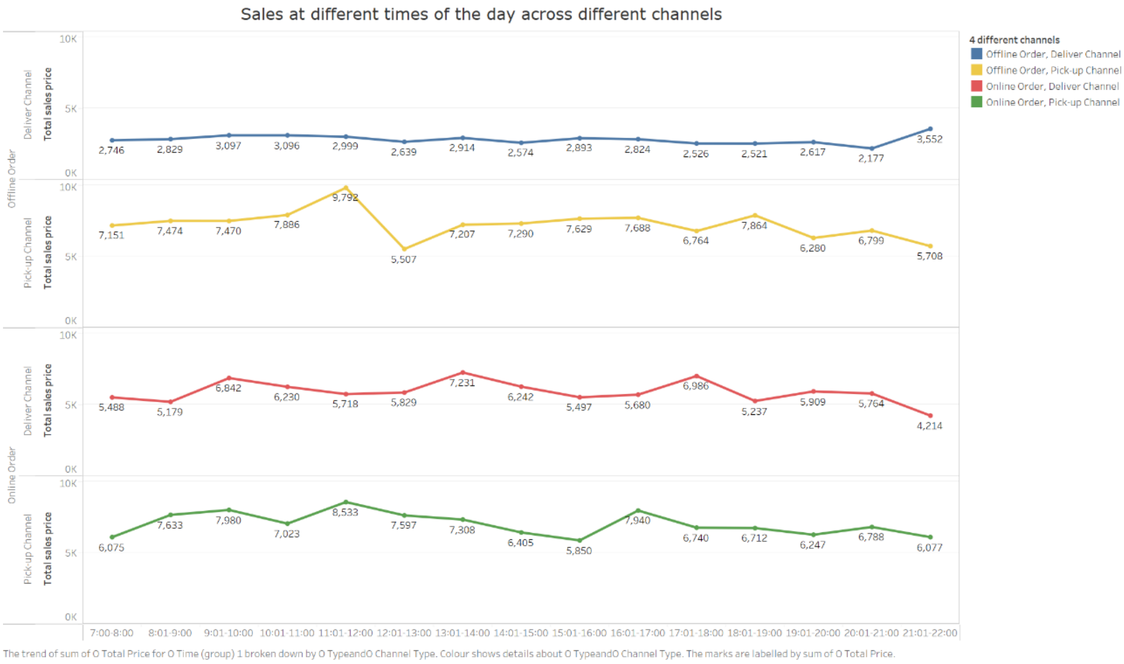


Fig5.11

6. Business Intelligence Analysis

1) Regression Analysis

I. Research Design

In the business intelligence analytics segment, we employ regression models for analysis. At first, we utilize control variables related to sales volume to establish a baseline model, which serves as a foundation for subsequently added variables. By comparing the baseline model with the modified model, improvements in the model after adding new variables can be assessed, ensuring the interpretability of the model and the intuitive understanding of the results:

$$Ln(Sale_i)=\alpha+\beta_1 Temp_i+\beta_2 Advertise_i+\beta_3 Discount_i+\beta_4 Rating_i+\varepsilon_i \quad (1)$$

Next, we introduce a dummy variable for store location to explore the sales volumes of products across different regions, examining whether there is a need for continued expansion or reduction of stores:

$$Ln(Sale_i)=\alpha+\beta_1 Temp_i+\beta_2 Advertise_i+\beta_3 Discount_i+\beta_4 Rating_i+\beta_5 Location_i+\varepsilon_i \quad (2)$$

Furthermore, building on Model (1), we add a dummy variable for channel to investigate the sales differences between online and offline channels, to determine whether there is a need to expand online channels:

$$Ln(Sale_i)=\alpha+\beta_1 Temp_i+\beta_2 Advertise_i+\beta_3 Discount_i+\beta_4 Rating_i+\beta_5 Channel_i+\varepsilon_i \quad (3)$$

Additionally, we incorporate a dummy variable for pre-made tea into Model 1 to explore the sales differences between pre-made tea and non-pre-made tea, assessing whether there is a need to continue developing this new product:

$$Ln(Sale_i)=\alpha+\beta_1 Temp_i+\beta_2 Advertise_i+\beta_3 Discount_i+\beta_4 Rating_i+\beta_5 Type_i+\varepsilon_i \quad (4)$$

Finally, we include all three aforementioned dummy variables into Model 1 along with interaction terms, to explore whether the moderating effects among these variables positively influence sales, potentially enhancing sales volumes significantly:

$$Ln(Sale_i)=\alpha+\beta_1 Temp_i+\beta_2 Advertise_i+\beta_3 Discount_i+\beta_4 Rating_i+\beta_5 Location_i+\beta_6 Channel_i+\beta_7 Type_i +\beta_8 (Location*Channel)_i+\beta_9 (Location*Type)_i+\beta_{10} (Type* Channel)_i+ \varepsilon_i \quad (5)$$

Overall, we excluded some missing data and collected detailed information on 3,000 beverage stores. Due to memory constraints, only one day's data is available in the database. To ensure the accuracy of the regression analysis results, we incorporate historical data from the previous year, including sales, advertising expenditures and other quantitative variables, to assist the model's analysis process. Below is a display and brief explanation of all the variables.

Control Variable Scale

Variable Name	Definition	Symbol
Sales of Products	The value of goods sold in a given period.	***Sale***
Temperature	The degree of heat in an environment.	***Temp***
Advertisement Input	The budget for marketing activities.	***Advertise***
Discount Rate	Reductions applied to the original selling price of goods	***Discount***
Rating of Products	Customer feedback on the product (rating 1-5).	***Rating***
Location of store	Location is divided into first-tier & second-tier cities (represent 1) and others (represents 0).	***Location***
Distribution Channel	The methods of selling products include both online (represents 1) and offline (represents 0).	***Channel***
Product Type	Products are mainly divided into pre-made tea (represents 1) and non-pre-made tea (represents 0).	***Type***

Fig6.1

II. Analysis of Empirical Results

At first, we conduct a descriptive statistical analysis to elucidate the characteristics and relationships within the data (Fig6.2).

The sales data from 3,000 observations shows an average of 146k rmb, with values ranging from 26k to 532k rmb and a low standard deviation of 0.88. Other averages include a temperature of 14 degrees, 45k rmb in advertising, a 30% discount, and high customer ratings of 4.3/6, indicating satisfaction and stability for model control.

Descriptive Statistics

	Count	Mean	Min	Max	Std
Sale	3000	146	26	532	0.88
Temp	3000	14	4	34	0.97
Advertise	3000	45	12	108	2.81
Discount	3000	0.3	0	0.5	2.24
Rating	3000	4.3	1	6	3.74

Fig6.2

Subsequently, we examine the linear relationships between each control variable to prevent the occurrence of collinearity or multicollinearity, which could introduce bias and affect the experimental outcomes (Fig6.3).

From the table, we can observe that the correlations between variables are all below 40%, demonstrating the absence of collinearity within the regression model, thus negating the need for adjustment.

Correlation Analysis

	Sale	Temp	Advertise	Discount	Rating
Sale	1				
Temp	0.224	1			
Advertise	0.134	0.324	1		
Discount	0.322	0.129	0.142	1	
Rating	0.188	0.274	0.323	0.135	1

Fig6.3

Next, we commence the construction of the formal regression model by first establishing a baseline model (Fig6.4).

The regression results of Model 1 indicate that temperature, advertising expenditure, discount magnitude, and customer ratings significantly predict sales, all with positive impacts. Despite being robust, the model's low adjusted R-squared of 0.35 suggests other significant variables might be missing, indicating the need for further model expansion.

Model 1 Regression Result

Dep. Varable: Ln*Sale*				
Explanatory Variables	coef	std err	t	P>|t|
Temp	0.15	5.49	1.95	0.04
Advertise	0.29	3.97	2.16	0.03
Discount	0.17	0.43	3.20	0.00
Rating	0.49	3.24	1.75	0.03
Obs	3000			
R-squared	0.33			
Adj. R-squared	0.35			

Fig6.4

Subsequently, we introduce the dummy variable *Location* into the model (Fig.5). In *Location*, we designate first and second-tier cities with a value of 1, while third-tier and other cities are assigned a value of 0.

From the table, we observe that the coefficient for *Location* is 0.07, suggesting that product sales in first and second-tier cities are, on average, only 7% higher than those in third-tier and other cities. Additionally, this result is statistically significant at the 5% level.

Moreover, upon the introduction of the regional dummy variable, there is an appreciable increase in the adjusted R-squared value, further validating the relationship between *Location* and *Sale*.

Our store expansion has primarily focused on first- and second-tier cities, yet sales in these cities are lower than in third-tier and other smaller cities. This indicates a problem with our expansion strategy. We can infer that although first- and second-tier cities have higher GDP and stronger consumer purchasing power, our mid-to-low-end products cannot attract significant consumer interest. Additionally, the influx of competing products has led to intense market competition as well, fragmenting the market share. Therefore, based on the analysis, we can reduce stores in first and second-tier cities and shift resources appropriately to third-tier and other cities, achieving cost reduction and efficiency gains.

Model 2 Regression Result

Dep. Varable: Ln*Sale*				
Explanatory Variables	coef	std err	t	P>|t|
Location	0.07	3.49	1.94	0.02
Temp	0.16	5.43	1.95	0.04
Advertise	0.34	3.97	2.16	0.03
Discount	0.22	0.43	3.20	0.00
Rating	0.52	3.24	1.75	0.03
Obs	3000			
R-squared	0.42			
Adj. R-squared	0.44			

Fig6.5

Moving forward, we augment the baseline model by incorporating *Channel* as a dummy variable (Fig6.6). Within this variable, online channels are coded as 1, and offline channels as 0.

The results from the table reveal that the coefficient for channel is 0.26, indicating that, on average, sales through online channels are 26% higher than those through offline channels, and this result is significant at the 1% level.

Furthermore, the adjusted R-squared value has also increased in this model, suggesting that the introduction of the channel variable is appropriate.

The online channel was recently launched, with significantly lower coverage than the offline channel, but its sales growth has already surpassed that of the offline channel. This strategy of expanding the online channel aligns with the public's preferences, as the proliferation of electronic devices has facilitated convenient ordering. Therefore, based on the analysis, we can conclude that the company should broaden its channels, shifting focus toward online sales, rationally allocate resources, and create higher-quality online services to achieve greater sales growth.

Model 3 Regression Result

Dep. Varable: Ln*Sale*				
Explanatory Variables	coef	std err	t	P>|t|
Channel	0.26	1.47	2.02	0.00
Temp	0.17	4.49	1.55	0.04
Advertise	0.34	3.67	2.56	0.03
Discount	0.21	0.43	3.23	0.00
Rating	0.52	3.24	1.76	0.03
Obs	3000			
R-squared	0.43			
Adj. R-squared	0.47			

Fig6.6

Additionally, we have introduced a dummy variable *Type* into the baseline model (Fig6.7). Pre-made tea is coded as 1, while non-pre-made tea is set to 0.

From the table, we note that the coefficient for *Type* is 0.2, indicating that sales growth of pre-made tea are 20% higher than those of non-pre-made tea, with this result being statistically significant at the 1% level.

Similarly, the adjusted R-squared of this model shows a certain increase compared to the baseline model, affirming the rationale for including *Type*.

From the analysis, it is evident that as a new product launched by the company, pre-made tea has rapidly gained customer favor and substantial purchases, further validating the company’s decision to introduce this product line. The company should, therefore, continue to diversify its pre-made tea offerings, meet customer demands, and foster sales growth.

Model 4 Regression Result

Dep. Varable: Ln*Sale*				
Explanatory Variables	coef	std err	t	P>|t|
Type	0.20	1.22	1.45	0.00
Temp	0.16	3.45	1.64	0.04
Advertise	0.37	2.67	2.56	0.03
Discount	0.25	0.43	3.25	0.00
Rating	0.54	3.26	1.76	0.03
Obs	3000			
R-squared	0.42			
Adj. R-squared	0.43			

Fig6.7

At last, Fig6.8 reveals that the coefficients for all three interaction variables are positive and larger than 0.3, indicating that they exert a mutually reinforcing influence on each other instead of eroding other's positive effect, and this result is statistically significant at the 1% level.

Moreover, it is observed that adjusted R-squared has further increased upon the basis of previous models, justifying the inclusion of three interaction variables.

During the model 2, model 3 and model 4, we confirmed the impact of these three dummy variables as individual factors on sales. However, store managers are concerned about the potential interactions between the three strategies. For instance, some speculate that the introduction of pre-made tea might accelerate product completion speed, making customers more inclined to order and collect in-store, thereby eroding the advantages of the online channel. Additionally, reducing the number of stores could impact sales of pre-made tea and hinder the expansion of the online channel, among other concerns. In this model, we further gain the insights that if the company concurrently downsizes stores, expands online channels, and vigorously develops pre-made tea products, there will be a more pronounced enhancement in sales, surpassing the cumulative positive impact of each measure taken individually. Therefore, we can conclude that the it is a justified and promising approach for company to continue to implement these three strategies to achieve a significant increase in sales.

Model 5 Regression Result

Dep. Varable: Ln*Sale*				
Explanatory Variables	coef	std err	t	P>|t|
Location	0.15	5.49	1.95	0.01
Channel	0.36	3.33	4.32	0.00
Type	0.28	2.45	5.32	0.00
Temp	0.23	3.43	1.43	0.02
Advertise	0.40	2.35	2.52	0.03
Discount	0.29	0.42	3.63	0.00
Rating	0.58	3.52	1.44	0.02
Interaction Variables				
Location:Channel	0.34	3.25	5.55	0.01
Location:Type	0.35	1.23	4.56	0.00
Channel:Type	0.46	2.42	7.32	0.01
Obs	3000			
R-squared	0.48			
Adj. R-squared	0.51			

Fig6.8

2) Forecasting Analysis

Based on the regression analysis conducted, we have gained insights into how the three main factors influence sales. To obtain a more accurate forecast for the future, we will proceed with predictive analysis using classical time series models. By leveraging historical data from the past year, we aim to predict the data for the upcoming year and verify whether the implementation of these three improvement measures will result in a steady increase in future sales.

In predictive analysis, we will use Auto Regressive Integrated Moving Average (ARIMA) model. The ARIMA model is a sophisticated tool for time series forecasting, which combines auto-regressive components, differencing for stationarity, and moving average elements, allowing it to make accurate predictions.

Firstly, we import the date and corresponding sales data, proceed with the auto-correlation and partial auto-correlation tests.

By examining the auto-correlation function (ACF) and partial auto-correlation (PACF) plots, we are able to further determine the values of p (the order of the AR) and q (the order of the MA). Upon examination, it is found that when p equals 2 and q equals 1, the Akaike Information Criterion (AIC) attains its minimum.

A p-value of 2 indicates that the sales data has a significant linear relationship with its values from the previous two years. This suggests that current sales are influenced not only by last year's sales but also by the sales from the year before. Such dependency reflect the sustained impact of recent store reform measures on consumer purchasing behavior and market activities. A q-value of 1 indicates that the model needs to consider the error term from the previous year to help predict future sales. This highlights that fluctuations in sales data are significantly influenced by internal variations over the past year, which indirectly corroborates the evident effects of store reductions, channel expansion, and the launch of pre-made teas on sales.

Based on the above mentioned analysis, we can propose the optimal ARIMA model.

ARIMA Regression Result

Sample: 2023.3-2024.3			Log Likelihood:-657.24	
Wald Chi2(5):61235.42			Prob>Chi2:0.00	
Explanatory Variables	coef	std err	z	P>|z|
_cons	7.20	1.22	4.35	0.00
AR:L1	1.28	0.06	-16.44	0.00
AR:L2	-0.94	2.67	2.56	0.00
MA:L1	-1.75	0.12	-16.23	0.00
MA:L2	0.79	0.15	5.42	0.00
MA:L3	-0.04	0.08	-0.62	0.00
Obs	3000			
Sigma	14.02			

Fig. 6.9

The Wald chi-square statistic is significant (chi = 61235.42 with p = 0.00), indicating that the overall model is statistically significant. The p value of AR and MR terms are all 0.00, suggesting that past sale values and errors are useful predictors for the current value. This implies that the current input of sales and date data can be quite perfectly aligned with the ARIMA model, allowing for relatively clear and accurate prediction outcomes. Thus, the explanatory variables, as a whole, significantly predict sales.

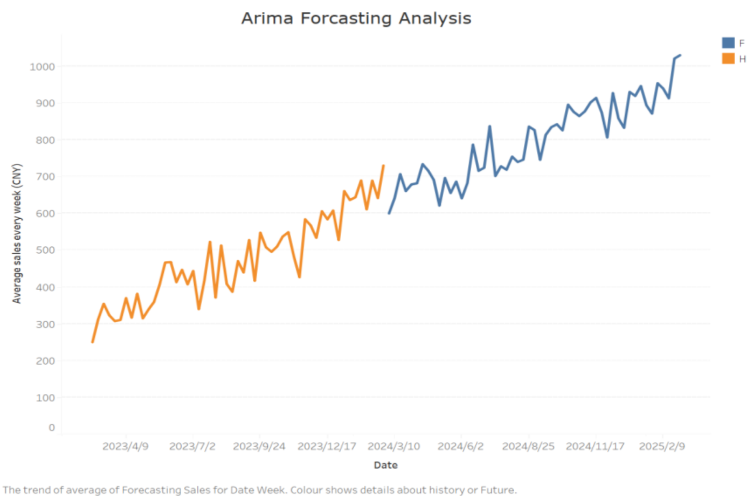


Fig. 6.10

Finally, we visualize the results of the ARIMA model (Fig6.10), projecting the sales levels for the upcoming year. The line graph reveals that the overall trend is one of tumultuous ascent. The forecasted sales outcomes derived from this predictive model align with our expectations and corroborate the results obtained from the previous regression model, namely that these three strategic enhancements will reduce costs, increase efficiency, attract more customers, and ultimately boost sales. However, some abnormal low values in the chart may be due to obstacles encountered during the launch of new pre-made tea products, server crashes affecting the online channel, or temporary drops in consumption caused by store reductions. Therefore, stores should regularly review and update these three innovative measures to ensure steady sales growth in the future.

7. Reflection on database solution

The database is designed to enhance data analysis capabilities for Notting's Taste across three dimensions to improve the company's profitability. Firstly, it constructs a third-party evaluation system to assist the company in selecting suitable third-party collaborators. The database combines customer and store ratings with the company's service pricing to help select the most suitable third-party companies from over 200 suppliers. The system initially establishes a data model with store and consumer ratings to evaluate service quality, record delivery times to store and customer addresses, and the final transaction price. Additionally, third parties can access information compiled specifically for them in the database. This allows third parties to adjust their services to offer better service based on their ratings and Notting's Taste can directly select the most suitable third parties. This evaluation system only rates the services of the suppliers themselves. However, Notting's Taste cannot rely entirely on this system, as different suppliers have different company scales; some may provide high-quality services but may not be able to handle Notting's Taste's large volume of business entirely. Therefore, when selecting third-party collaborators using the database, Notting's Taste requires considering human consultation and management negotiation to determine different scales or contracts.

Secondly, the database establishes comprehensive data statistics for each store, dynamically managing the store network. Specifically, the company can use the database to evaluate which stores should be optimized based on each store's order information, inventory levels, and employee recruitment and departure times. Additionally, by setting thresholds (minimum and maximum inventory levels) through SQL, users can track inventory changes using SQL queries to restock or reduce purchases in a timely manner.

The database also analyzes sales differences between stores in different regions in-depth. This allows for determining which stores need more investment, which may need to reduce investment or even close to optimize the profit structure. Currently, the company is in the early stages of expansion and particularly needs a detailed analysis of business and inventory data for different regional stores, as well as a detailed analysis of online and offline order data to ensure that every investment decision is based on real-time accurate data analysis to support dynamic store management. However, the database is limited to analyzing the profitability between product costs and revenues to achieve dynamic store expansion. It cannot store investment analysis for products other than products, such as rent, decoration, and local taxes, so more financial data needs to be integrated for strategic management. Additionally, the values set for the SQL queries for minimum or maximum inventory levels are subjective and may be influenced by regional factors or other factors, so the company needs to rationally analyze these extreme cases to improve the inventory management system.

Thirdly, the database will conduct an in-depth analysis of the product structure, as products are the core of the company's competitiveness in the market, directly affecting market acceptance and competitive positioning. By deeply analyzing the product components and acceptance of different sales channels in the database, the company can more accurately determine which products perform well, which products need improvement or discontinuation. Specifically, the company can use the database to analyze product sales and cost data, such as raw material costs and selling prices, evaluate different online and offline sales scenarios, and store locations to adapt to market changes and consumer demands. With this data, the company can optimize raw material distribution, focus on profitable, high-sales products, while reducing investment in low-profit products. Additionally, the company can accurately assess how to invest in different products in different regions, adjust product portfolios based on sales and consumer preferences, such as regional products. By analyzing product discounts, it seeks to explore price elasticity to find a supply-demand balance. This in-depth analysis not only helps Notting's Taste maintain a leading position in the market competition but also ensures that every product decision is supported by accurate and comprehensive data. However, relying solely on historical data for strategic decisions may not accurately predict future market trends, so the company also needs to supplement database analysis with market research and case studies to provide a broader perspective to guide decision-making.

Appendix

Appendix1: Meeting Record

2023-24 Minutes of the meeting			
Date	2024.3.28	**会议时间**	19:00~ 20:00
Location	Library	**会议记录**	Renyu Jiang
Attendance	Renyu Jiang, Yuxiao Deng, Ziyu Liu, Zhengyi Lin, Lanshun Yuan		
Request for Leave	/		
Absence	/		
Content of Meeting			

Meeting Notes: Initial Discussion on Database Implementation

- Team Roles and Responsibilities:

  Established clear roles for each team member.

  Conducted a task analysis to pinpoint the critical aspects of the project.

  Yuxiao Deng:ER diagram, Data visualization, Data

  Ziyu Liu: Data dictionary: System definition

  Zhengyi Lin: SQL, Data

  Lanshun Yuan: Business intelligence analysis

  Renyu Jiang: Background, Reflection, Report structure
- Project Framework:

Created a broad framework focused on analyzing company profitability and expansion problem.

- Task Distribution:

  Clarified the distribution of tasks among team members.

  Discussed the distinctions between different tasks.

  Identified issues and prepared to attend office hours for further guidance.

- Focus:

  focus should be on the operation of the store and on data analysis.

2023-24 Minutes of the meeting			
Date	2024.4.4	**会议时间**	19:00~ 20:00
Location	Library	**会议记录**	Renyu Jiang
Attendance	Renyu Jiang, Yuxiao Deng, Ziyu Liu, Zhengyi Lin, Lanshun Yuan		
Request for Leave	/		
Absence	/		

Content of Meeting

Discussion

- Data Visualization for Papers: The paper should include graphs and a brief text explanation.
- Purpose of SQL: Explain what SQL is used for, who uses it, the goals it achieves, and include some code demonstrations.
- Displaying ER Diagrams: Show a clear, high-resolution ER diagram that may include intersecting relationships.
- Issues with Relationship Attributes: Relationship attributes are not entities. Review the PowerPoint to understand how attributes within relationships are transformed.
- Necessity of Distinguishing Strong and Weak Relationships: It is unnecessary.

  Chain Store Quantities: Any number is acceptable; the amount of data is not graded but must be reasonable.

BG:

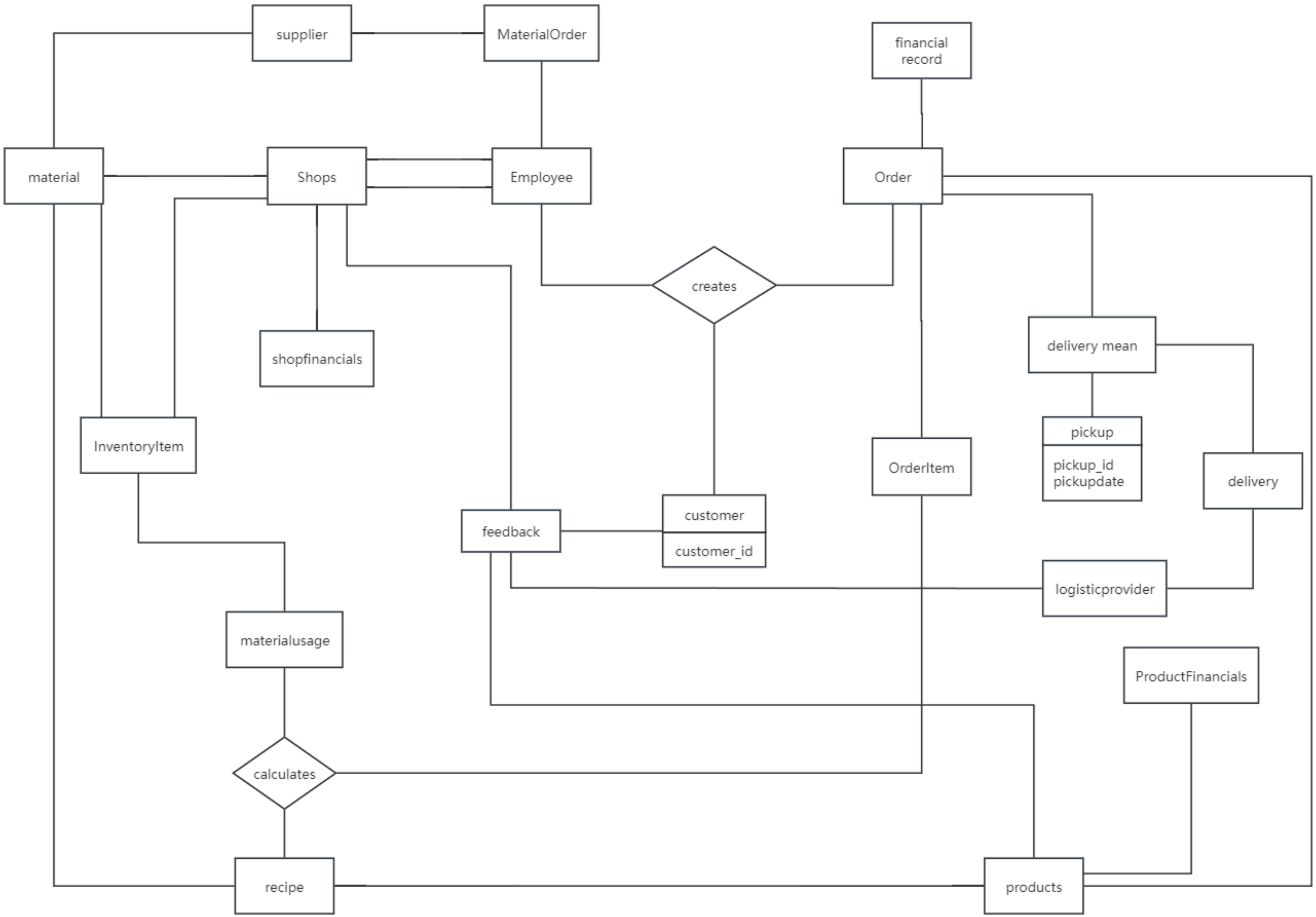
In discussing the number of new franchise stores, it's crucial to also focus on the closure rate. The relationship between these figures is a key indicator of the health of the franchise system. Notting's taste emphasizes a small store model and a takeout business, which can significantly reduce costs related to rent, decoration, and staffing, while also supporting delivery services and pickup windows, enhancing operational efficiency. This cost-effective small-store model is highly attractive to franchisees, who should expand dynamically based on this strategy.

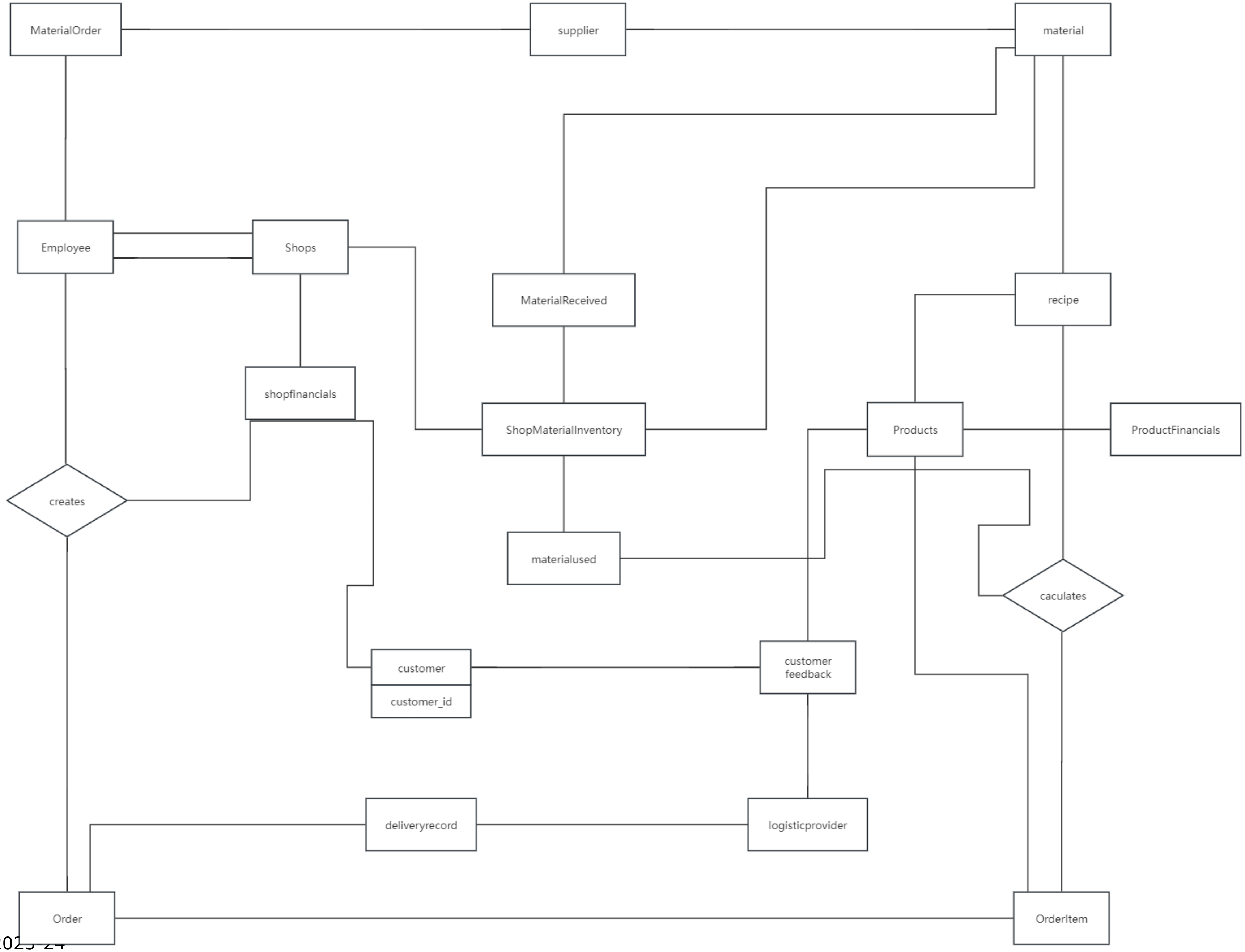
2023-24 Minutes of the meeting			
Date	2024.4.11	**会议时间**	19:00~ 20:00
Location	Library	**会议记录**	Renyu Jiang
Attendance	Renyu Jiang, Yuxiao Deng, Ziyu Liu, Zhengyi Lin, Lanshun Yuan		
Request for Leave	/		
Absence	/		
Content of Meeting			

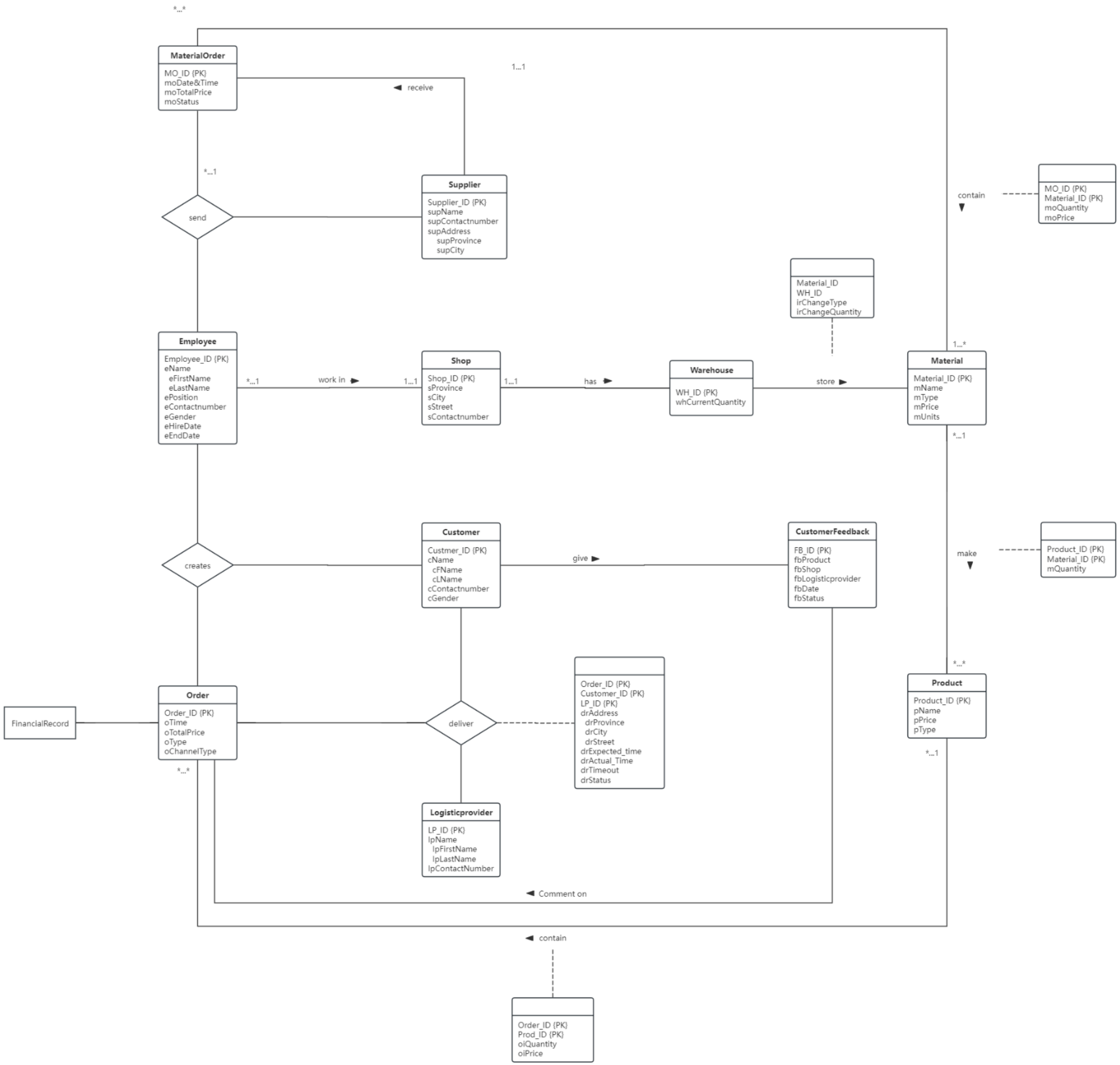
Discussion

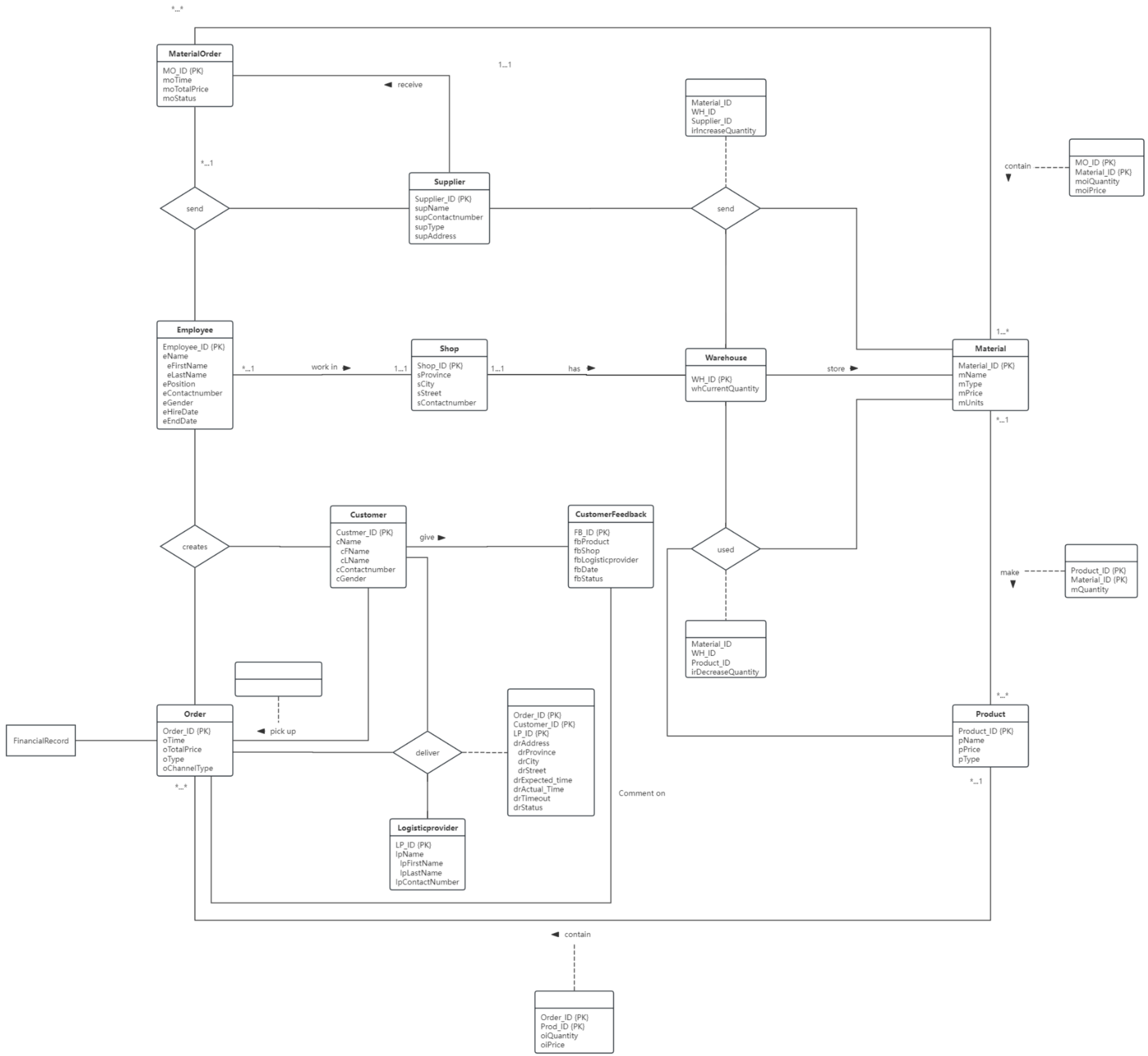
- Attributes in ER Diagrams: The diagram should detail attributes as much as possible; primary keys need to be included but other keys are not necessary.
- Connections in ER Diagrams: Only connected entities have foreign keys. Primary and foreign keys are connected; ensure relationships are well defined and check them again.
- Entity and Foreign Keys: Entities should not include foreign keys repeatedly; foreign keys should be distinctly marked.
- Combining Different Tables to Represent Suppliers: There's no need for a separate table for suppliers in the ER diagram.

Appendix2: Historical ER diagram




MaterialOrder
supplier
material
Employee
Shops
MaterialReceived
recipe
shopfinancials
ShopMaterialInventory
Products
ProductFinancials
creates
materialused
caculates
customer
customer_id
customer
feedback
deliveryrecord
logisticprovider
Order
OrderItem


MaterialOrder
MO_ID (PK)
moDate&Time
moTotalPrice
moStatus
receive
1...1
...
*...1
send
Supplier
Supplier_ID (PK)
supName
supContactnumber
supAddress
supProvince
supCity
contain
MO_ID (PK)
Material_ID (PK)
moQuantity
moPrice
Material_ID
WH_ID
irChangeType
irChangeQuantity
Employee
Employee_ID (PK)
eName
eFirstName
eLastName
ePosition
eContactnumber
eGender
eHireDate
eEndDate
*...1
work in
1...1
Shop
Shop_ID (PK)
sProvince
sCity
sStreet
sContactnumber
1...1
has
Warehouse
WH_ID (PK)
whCurrentQuantity
store
1...*
Material
Material_ID (PK)
mName
mType
mPrice
mUnits
*...1
creates
Customer
Custmer_ID (PK)
cName
cFName
cLName
cContactnumber
cGender
give
CustomerFeedback
FB_ID (PK)
fbProduct
fbShop
fbLogisticprovider
fbDate
fbStatus
make
Product_ID (PK)
Material_ID (PK)
mQuantity
...
Product
Product_ID (PK)
pName
pPrice
pType
*...1
Order_ID (PK)
Customer_ID (PK)
LP_ID (PK)
drAddress
drProvince
drCity
drStreet
drExpected_time
drActual_Time
drTimeout
drStatus
FinancialRecord
Order
Order_ID (PK)
oTime
oTotalPrice
oType
oChannelType
...
deliver
Logisticprovider
LP_ID (PK)
lpName
lpFirstName
lpLastName
lpContactNumber
Comment on
contain
Order_ID (PK)
Prod_ID (PK)
oiQuantity
oiPrice



...
MaterialOrder
MO_ID (PK)
moTime
moTotalPrice
moStatus
receive
1...1
Material_ID
WH_ID
Supplier_ID
irIncreaseQuantity
contain
MO_ID (PK)
Material_ID (PK)
moiQuantity
moiPrice
*...1
send
Supplier
Supplier_ID (PK)
supName
supContactnumber
supType
supAddress
send
Employee
Employee_ID (PK)
eName
eFirstName
eLastName
ePosition
eContactnumber
eGender
eHireDate
eEndDate
*...1
work in
1...1
Shop
Shop_ID (PK)
sProvince
sCity
sStreet
sContactnumber
1...1
has
Warehouse
WH_ID (PK)
whCurrentQuantity
store
1...*
Material
Material_ID (PK)
mName
mType
mPrice
mUnits
*...1
creates
Customer
Custmer_ID (PK)
cName
cFName
cLName
cContactnumber
cGender
give
CustomerFeedback
FB_ID (PK)
fbProduct
fbShop
fbLogisticprovider
fbDate
fbStatus
used
make
Product_ID (PK)
Material_ID (PK)
mQuantity
Material_ID
WH_ID
Product_ID
irDecreaseQuantity
FinancialRecord
Order
Order_ID (PK)
oTime
oTotalPrice
oType
oChannelType
pick up
deliver
Order_ID (PK)
Customer_ID (PK)
LP_ID (PK)
drAddress
drProvince
drCity
drStreet
drExpected_time
drActual_Time
drTimeout
drStatus
Comment on
...
Product
Product_ID (PK)
pName
pPrice
pType
*...1
...
Logisticprovider
LP_ID (PK)
lpName
lpFirstName
lpLastName
lpContactNumber
contain
Order_ID (PK)
Prod_ID (PK)
oiQuantity
oiPrice