Final Project of Relational Database Management

Alexia

1. development approach

There are mainly 3 phases in my development process which are requirement analysis, data modelling(conceptual, logical, physical) and normalisation .

1.1 requirement analysis

In the initial stage of the project, I conducted a thorough analysis of the requirements provided by the grocery outlet, and held interviews with its manager. Through this process, I elucidated the mission statement for the company's database, which aims to efficiently manage pertinent data related to its products and stakeholders, ensuring that the data remains current and accurate, facilitating seamless daily operations and supporting optional business decision-making. Furthermore, the mission objectives encompass the following key points:

- Maintain informations on all the employees
- Keep track of all the inventories
- Keep track of all the customers
- Keep track of all the sales
- Provide the information of the products and relevant services to customers

Drawing insights from the forms and documents provided by the company, I formulated both a primary field list and a calculation field list. These lists will serve as foundational elements for the upcoming phase, data modelling.

Preliminarily field list

(subject 1)Customer:	(subject 2)Product:	(subject 3)Supplier:
CustomerID(PK) Name Address, City, State, Zip Type Status ContactNumber Fax LastPurchase Notes	Product ID (PK) Category ProductDesign ProductVersion ProductDescription SRP *(suggested retail price)* CurrentQuantity	SupplierID (PK) CompanyName Address, City, State, Zip ContactName ContactNumber Status
(subject 4)Employee: EmployeeID (PK) EmployeeName	(subject 5)Invoice: SupplierID(FK) CustomerID(FK)	

DateHired Address, City, State, Zip ContactNumber Birthday SSN InstitutionName InstitutionAddress GraduatedYear	InvoiceNumber (PK) Date ProductID(FK) Quantity SRP *(suggested retail price)* EmployeeID(FK) HourlySalary Discount Terms Comments	

Calculated list

Customer: Age	Invoice: Amount	Employee information: Age CommitmentYear

1.2 data modelling

In this stage, three three types of data models are used to facilitate the database development process in SQL Server later.

The first one is a logical data model which is supposed to show the entity, attributes and relationship of the tables.

In this case, there are a total of five tables including Customer, Product, Supplier, Employee and Event. The relationship between the tables are explained below and presented on Figure 1.

- A customer can purchase multiple types of products; a type of products can be sold to many customers
- A supplier can supply multiple types of products; a type to products can be supplier by multiple suppliers
- An employee can serve multiple customers, vice versa
- An employee can sell multiple types of products, vice versa

A customer may purchase products from the grocery once or /multiple times; an invoice includes only one customer.

A product can appear in one or many invoice; vise versa

A supplier can supply one or more products in an invoice; an invoice can include one more more suppliers

A employee may have taken 1 or multiple Invoice; an invoice can include only 1 employees

	Customer	Product	Supplier	Employee	Invoice
Customer	N/A	M:N	N/A	M:N	1:1/N
Product		N/A	M:N	M:N	1/N : 1/N
Supplier			N/A	N/A	1/N : 1/N
Employee				N/A	1:1/N
Invoice					N/A

N/A: Not applicable, 1:0/N: 1 to 0 or many, M/N: Many to many

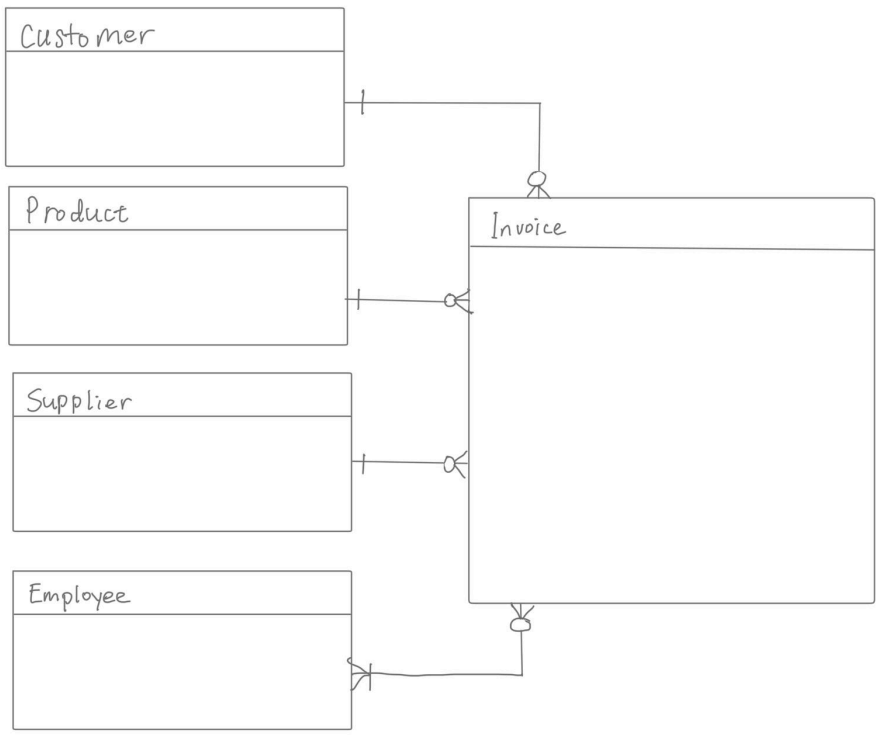

Figure1. Logical model

Based on the logical model, the conceptual model is developed. The primary key(PK), foreign key(FK) and the fields in each table of the database are displayed in Figure 2.

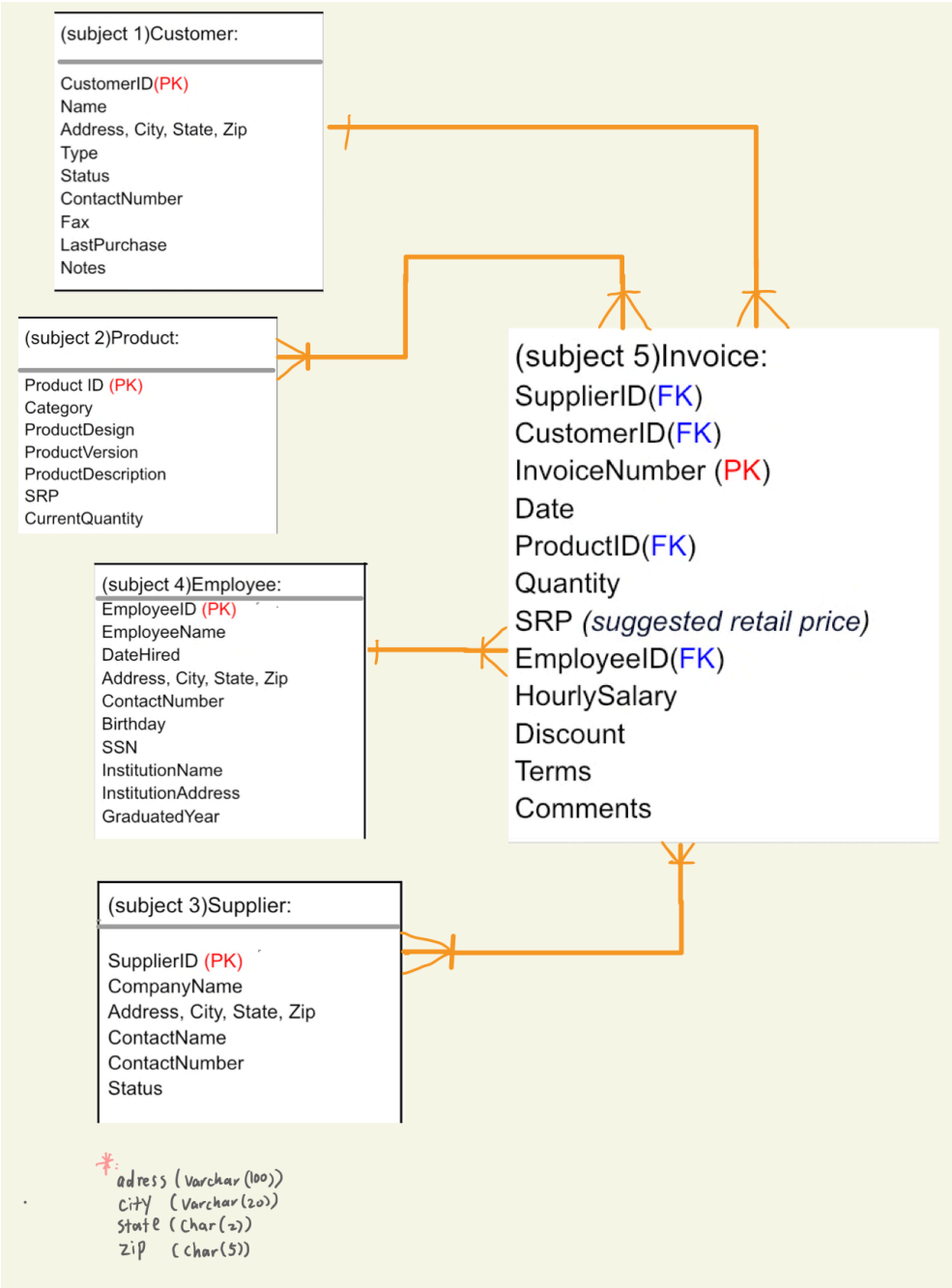

Figure 2. Conceptual model

Based on the conceptual model, a physical model (shown in Figure 3) is developed. It shows a more detailed and holistic view of the database.

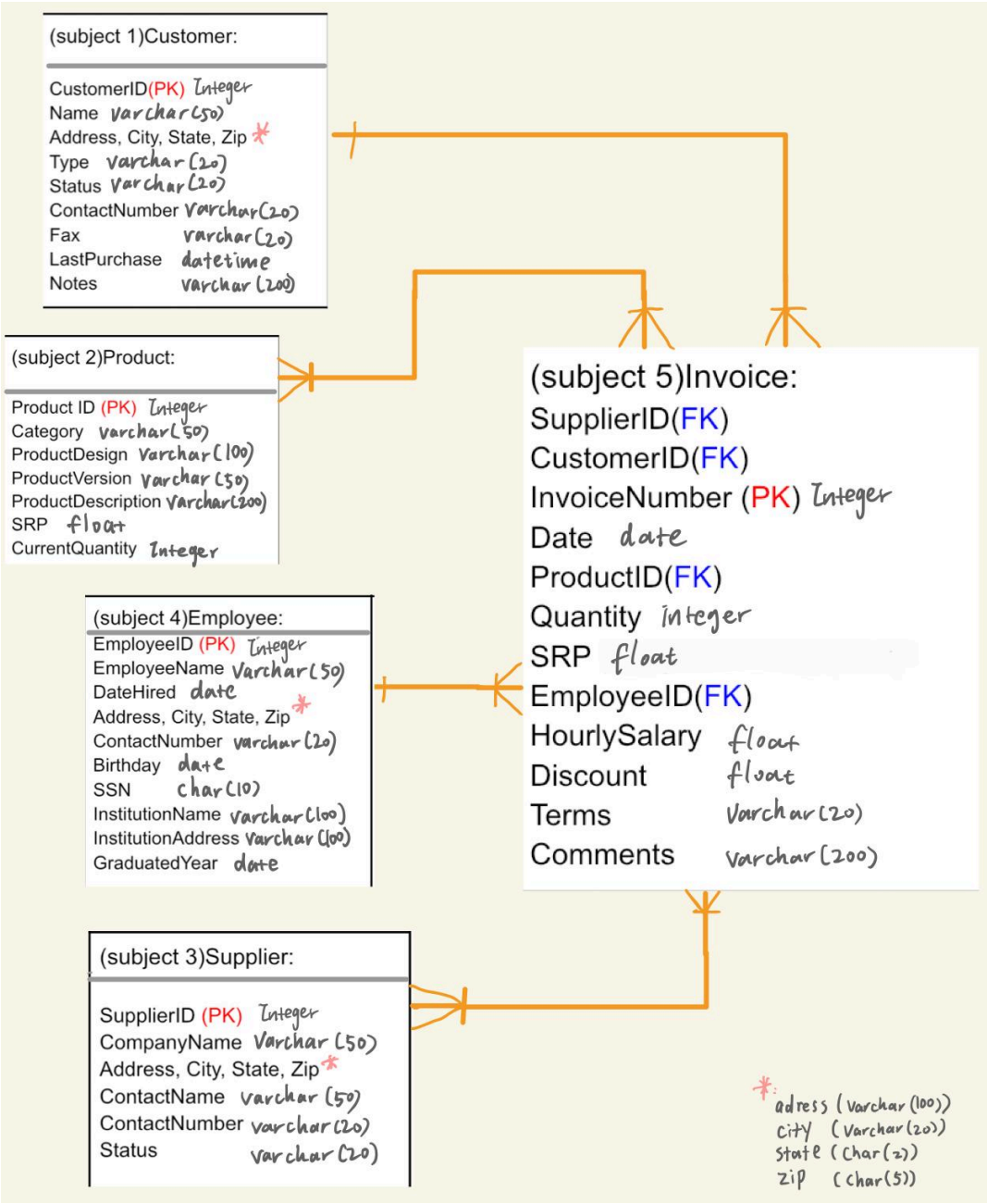

Figure3. physical model

However, this physical model is not ideal enough. Certain fields in a table are indirectly dependent such as HourlySalary and EmployeeID in the Invoice table. Moreover, the Address, Zip, State And City in some tables are slightly redundant. Hence, normalisation is required to modify physical model.

1.3 Normalisation and Business Rules

Normalisation is the applied on physical model to eliminate data redundancy and enhance data integrity in the table.

According to Normalisation Form #2 and #3 (refer to Principles of Design Process), the details of companies, products and customers are removed from the Invoice table. The comment is kept for allowing employees to input any other information regarding products or customers, for example, '25% discount due to slight scratch on product', 'Product04 is given as free gift (Christmas Offer)'.

Moreover, the Location table is created as a validation table to ensure accurate data input and less redundancy in the database.

Lastly, the business rules of Mike's Bicycle includes: 1. Only managers are eligible to earn a bonus of 5 or more percent of his/her salary. 2. Each employee is allowed to earn more than $100,000 or less than $30,000. Therefore, the table named Salary is created to ensure the business rules are implemented in the database system. It has a one-to-one relationship with the Employee table.

Therefore, the modified preliminary field table and physical table (Figure 4) are shown below:

Preliminarily field list (Modified)

Customer	Product	Supplier
CustomerID(PK) Name Address, Zip (FK) Type Status ContactNumber EmailAddress Fax LastPurchase Notes	Product ID (PK) Category ProductDesign ProductVersion ProductDescription SRP CurrentQuantity	SupplierID (PK) CompanyName ContactName ContactNumber Status EmailAddress Address, Zip (FK)
Employee	Invoice	Location

EmployeeID(PK) Name DateHired Address Zip (FK) ContactNumber Birthday SSN InstitutionName InstitutionAddress GraduatedYear	InvoiceID (PK) Date Discount Quantity Comments SupplierID(FK) CustomerID(FK) ProductID(FK) EmployeeID(FK)	Zip (PK) City State
Salary SalaryID (PK) EmployeeID (FK) Manager Bonus HourlySalary TotalSalary		

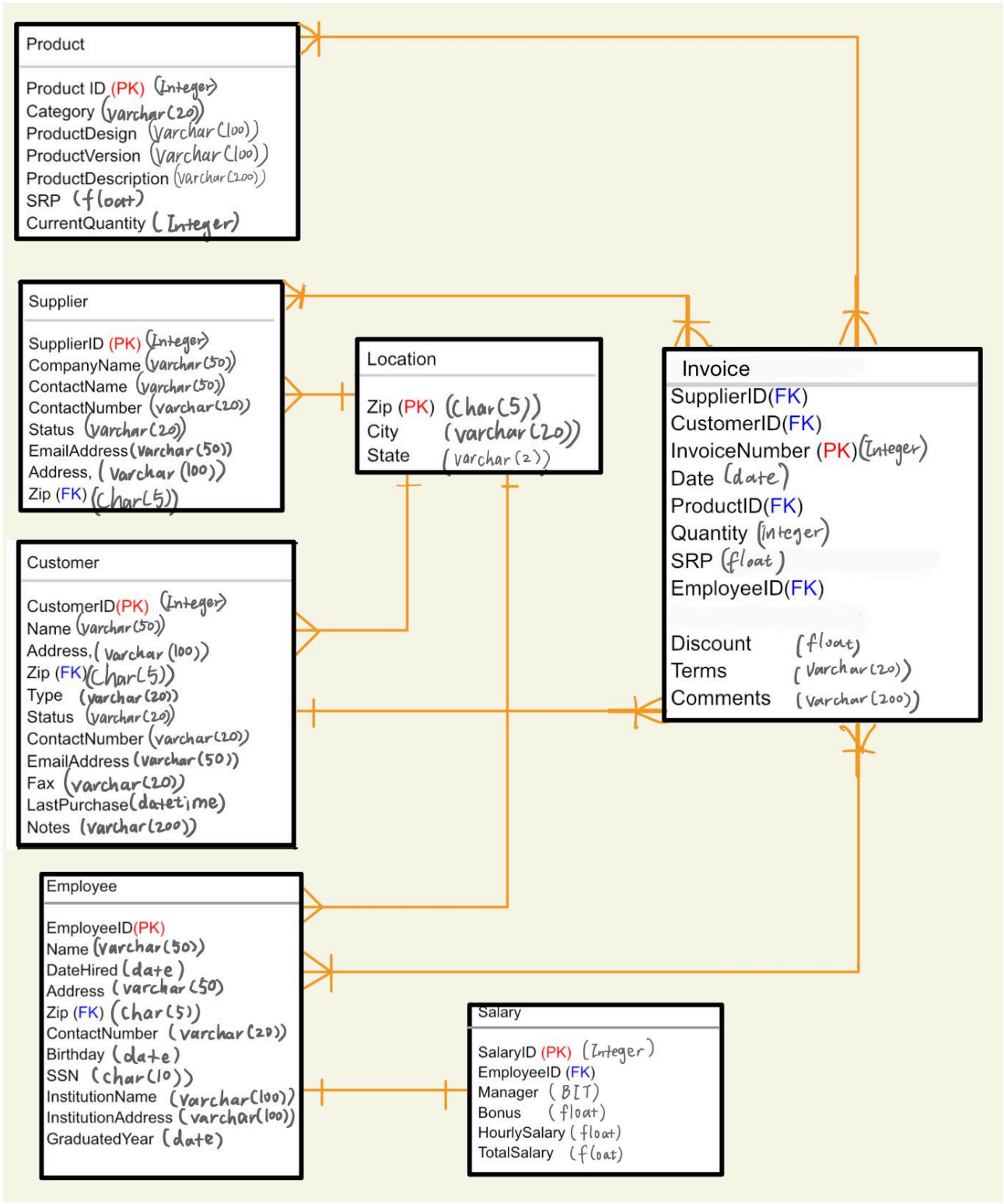

Figure4. Modified physical model

2. Principles of Design Process

- Use concise words to name fields and tables
- Avoid using punctuations in naming fields and tables
- Each table has a primary key
- Format the SQL codings properly to read and debug easily
- Normalisation:
  1.A single cell cannot hold multiple values.
  2.The table should not possess partial dependency.
  3.Non-prime attributes become directly (non-transitively) reliant on candidate keys
- Business rules:
  1. As part of our table definition, we'll enforce a company rule that no employee in our system is allowed to earn more than $100,000 or less than $30,000.
  2. Managers in our company can earn any bonus percentage but they are the only employee type that is allowed to earn a bonus of 5 or more percent of his/her salary

3. SQL codes

The sql codes provided cover the tasks below:

b. Create all tables normalized to third form

c. Add all appropriate columns with data types

d. Use Alter command to add all Primary and Foreign Keys

e. Insert sample data into the tables.

f. Write a basic SELECT statement for at least two table returning all or a select number of columns with sample data

g. Create constraint on employees implementing following business rules

- No employee in the database system is allowed to earn more than $100,000 or less than $30,000.
- Managers in Mike’s Bike can earn any bonus percentage but they are the only employee type that is allowed to earn a bonus of 5 or more percent of his/her salary

3.1 Create tables

GO

3.2 Add keys

-- connect Location table with Customer, Supplier and Employee

3.3 Insert values

--insert values into tables

3.4 Select samples

SELECT* FROM Employee

	EmployeeID	EmployeeName	DateHired	Address	Zip	ContactNumber	Birthday	SSN	InstitutionName	InstitutionAddress	Graduated
1	4	Emma	2008-01-01	S201 Bonbon Dr	12345	5113567633	1985-09-09	9876543210	UCB	University Avenue	2007-08-1
2	5	Ezra	2009-02-01	S201 Papillon Dr	12346	5119877633	1986-09-09	9876543211	UCSD	9500 Gilman Dr, La Jolla	2008-08-1
3	6	Ely	2010-03-01	S201 Bonsior Dr	12347	5116547633	1987-09-09	9876543212	UCSB	Santa Barbara,CA	2009-08-1

Figure5. Employee table

SELECT* FROM Location

	Zip	City	State
1	12345	Los Angeles	CA
2	12346	Berkeley	CA
3	12347	Sausilito	CA
4	12348	Alton	IL
5	12349	Aurora	IL
6	12350	Zion	IL
7	12351	Cocoa Beach	FL

Figure6. Location table

SELECT* FROM Supplier

	SupplierID	Address	Zip	EmailAddress	CompanyName	ContactName	ContactNumber	Status
1	3	S200 Reeves Dr	12345	al@gmail.com	Google	Alexia	5103567633	Active
2	4	S200 Bras Dr	12346	giwm@apple.com	Wholefood Market	Gina	+6583085858	Inactive
3	5	S200 Tete Dr	12347	sola@sony.com	La poche	Sophie	+15103567643	Active

Figure7. Supplier table

SELECT* FROM Customer

	CustomerID	Name	Address	Zip	EmailAddress	Type	Status	ContactNumber	Fax	LastPurchase	Notes
1	1	Cindy	S202 Le Chat Dr	12348	ci@outlook.com	VIP	Valid	6103567633	cin12135	2023-03-15 12:13:14.000	customer relationship m
2	2	Caleb	S202 Le chein Dr	12349	ca@outlook.com	Membersip	Valid	6113567633	calebfax123	2023-12-01 01:21:31.000	NA
3	3	Chloe	S202 Citron Dr	12350	ch@outlook.com	Free User	Valid	6203598933	chloefax4456	2020-07-15 21:45:51.000	NA
4	4	Calvin	S202 Citron Dr	12351	cv@outlook.com	Free User	Deleted	7103567633	cal90733	2022-12-15 17:13:00.000	NA

Figure8. Customer table

SELECT* FROM Salary

	SalaryID	EmployeeID	Manager	Bonus	HourlySalary	TotalSalary
1	1	4	1	0.55	100	50000
2	2	5	0	0.1	80	33000
3	3	6	0	0.25	90	45000

Figure9. Salary table

SELECT* FROM Product

	ProductID	Category	ProductDesign	ProductVersion	ProductDescription	SRP	CurrentQuantity
1	1	Electronic Device	Google Glass	2013.04 version	smart glass developed by Google X	599.99	10
2	2	Litchen Condiments	Brown Sugar	Wholefood Market	NA	9.09	100
3	3	Skincare	Face cream	La poche	Lotion for dry skin	89.99	50

Figure10. Product table

SELECT* FROM Invoice

	InvoiceID	Dates	Discount	Terms	Comments	SupplierID	CustomerID	ProductID	EmployeeID
1	3	2023-01-02 00:00:00.000	0.1	5	NA	3	1	1	4
2	4	2023-02-23 00:00:00.000	0.25	20	NA	4	2	2	5
3	5	2023-08-15 00:00:00.000	0	30	NA	5	3	3	6

Figure11. Invoice table

ER diagram

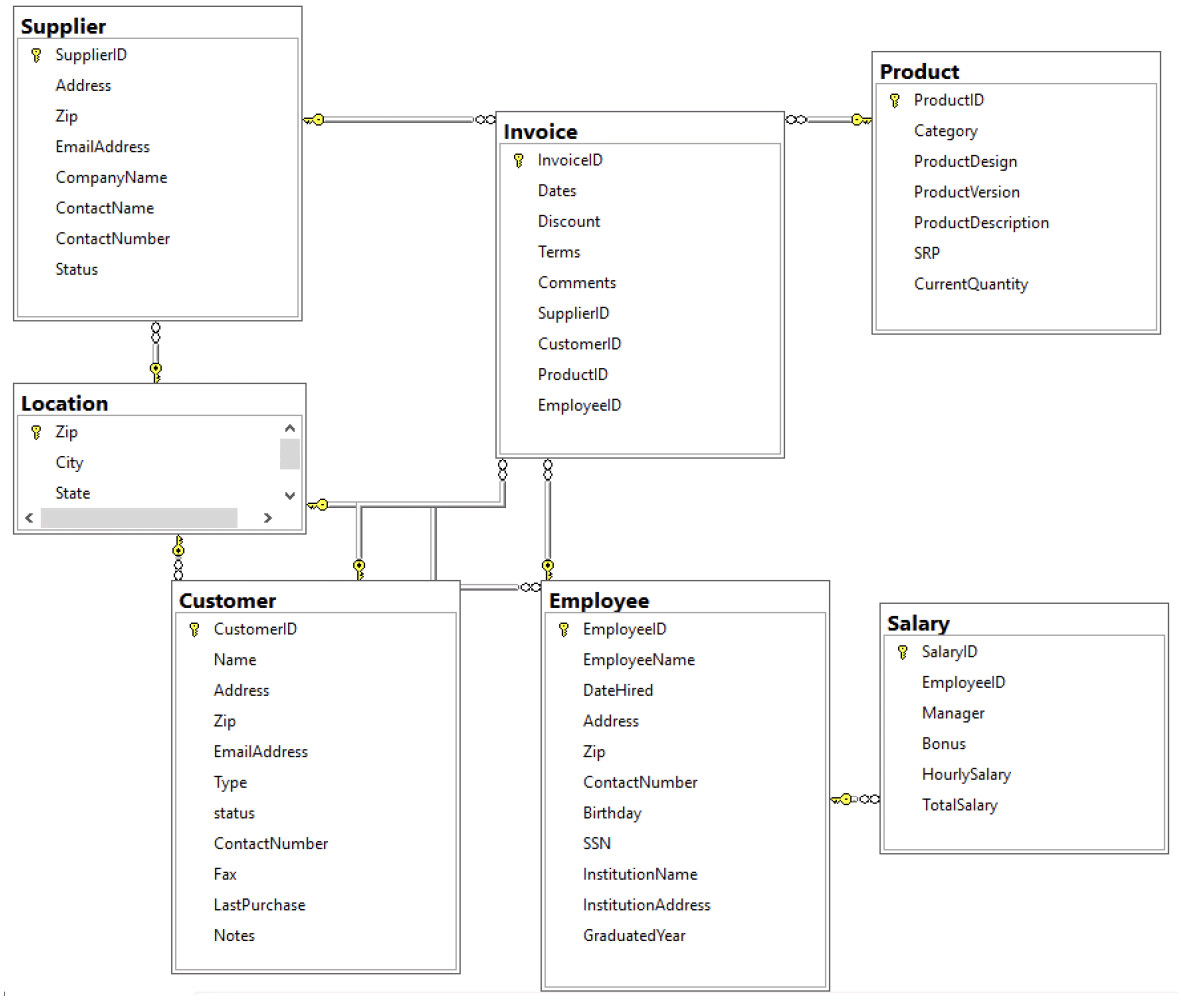

Figure12 ER Diagram (Column view)

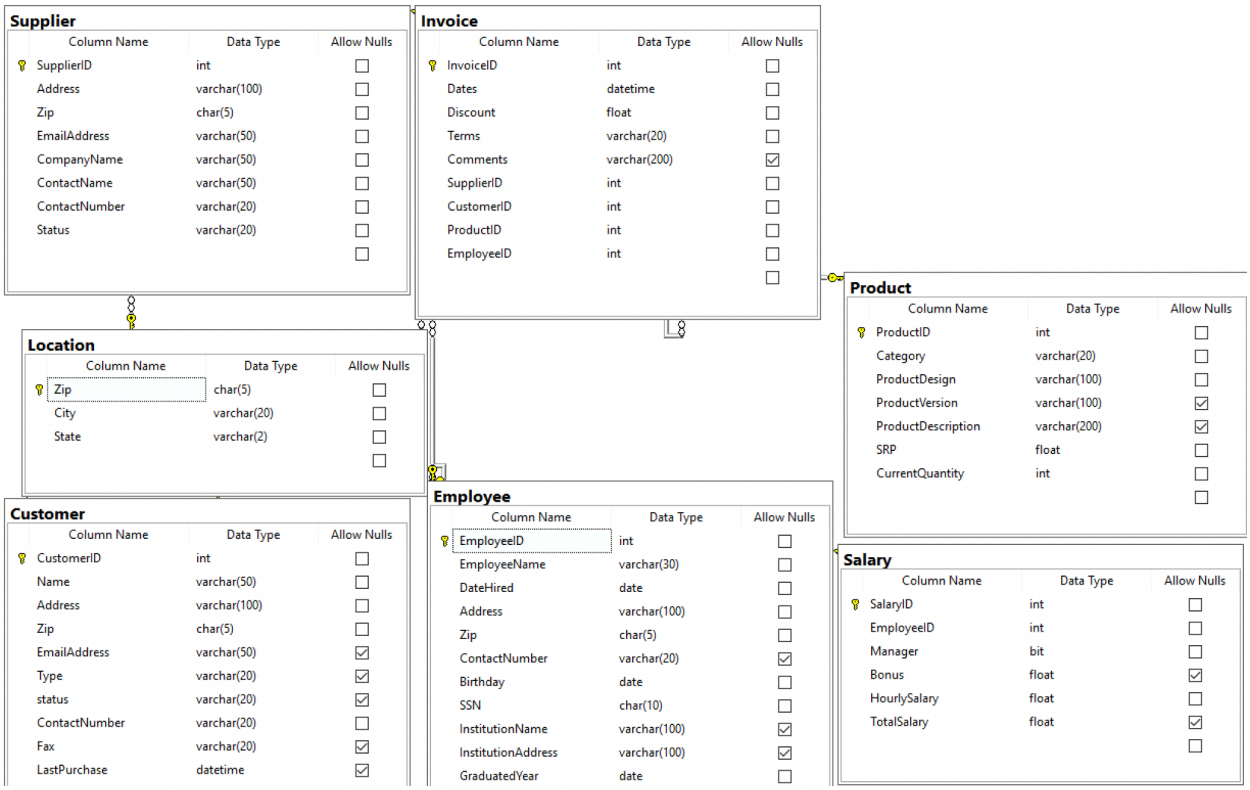

Figure13 ER Diagram (Standard view)