

To optimize in the process of searching and using data, I have designed a Database with 15 tables as shown above.

**As for merchant companies**

1. **A member of company** can suggest/propose to buy **products** of **other companies** and a department or company manager of the company can **review/ approve** the proposal of its staff ( its staff **cannot approve** )

When any member of the company proposes to buy something from other companies, we can check which company and department the member belongs to by relying on **merchantMemId foreign key** at the **order table** and easily determine who is the manager. In **merchant\_companies\_member table**, **role field** with int datatype (e.g: 0, 1, 2 is company manager, department manager, member respectively), will support for **authorization**.

At **orderDetail table**, have a relation many to one with **order table**, we will be able to check products from which companies by **merchantComId FK** at **products tables**

1. **Discounts** can **be applied for some products**, **company/department managers** can **update/create/delete discounts**

At **the discounts table**, rely on the **merchantComId** **foreign key** to identify the manager of the company and the relevant department. And based on the **discountId FK** at **products table**, we can know which products applied discounts

1. **Payment** info need to be recorded with all possible details

With **three tables as order, orderDetail and payments**, they will record all data in detail about time, product, quantity, buyer, buying company, etc.

**As for audit companies**

1. **Company managers** can **assign/allow their staff** to **view payments** of a **certain company**

In **audit\_companies\_members table**, **role field** with int datatype (e.g: 0, 1 is company manager, member respectively), will support for **authorization**.

**MerchantComId FK** at **the audit\_company\_members table** that allow staff is able to view payments of a certain merchant company, and **auditComId FK** will help determine who is its manager.

1. staff can view payments of other companies (if their managers allow them), but cannot update/delete/create

In **audit\_companies\_members table**, **role field** with int datatype (e.g: 0, 1 is company manager, member respectively), will support for **authorization**.