

note: please add a short description of the ERD so that we could understand what it means.

each employee has 0-1 account. but what about the clients?

* I think it’s not necessary to write the cardinality of the accounts of the clients since there can be no duplicate accounts. Also, account is for Clients and for Employee as well.

Isn’t employee supposed to be a subclass of account along with master acc and client acc?

* It’s already written that employee has an account meaning employee gets the same attributes plus the employee attributes.

employee and master account manages transactions while client accounts can order.

* What’s master account?
* Account is for Employee who manages the transactions and Clients who places an orders.

What is the difference of rental and order + orderline? doesn’t it have the same meaning? other than quantity.

* Customer requests for rental from the products.
* Customer places an order that adds to the orderLine (like shopping cart)

apparently there is supposed to be a forSale which was clarified by the analysts. My idea is that you separate orderline into 2 different classes for for Rent and for Sale since rental seems to be the same as order+orderline.

order > forRent

order > forSale

forRent > product

forSale > Product

transaction> order

forSale contains quantity and unit price. total price can be taken from queries.

forRent contains date returned

* But forSale and product has the same attribute.
* Yup, that’s why it’s written as +/totalPrice since it can generated.
* Rental has dateReturned attribute already.

order will become a superclass and forRent and forSale become a subclass since they both are a type of order.

* Up ^