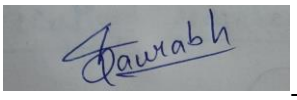
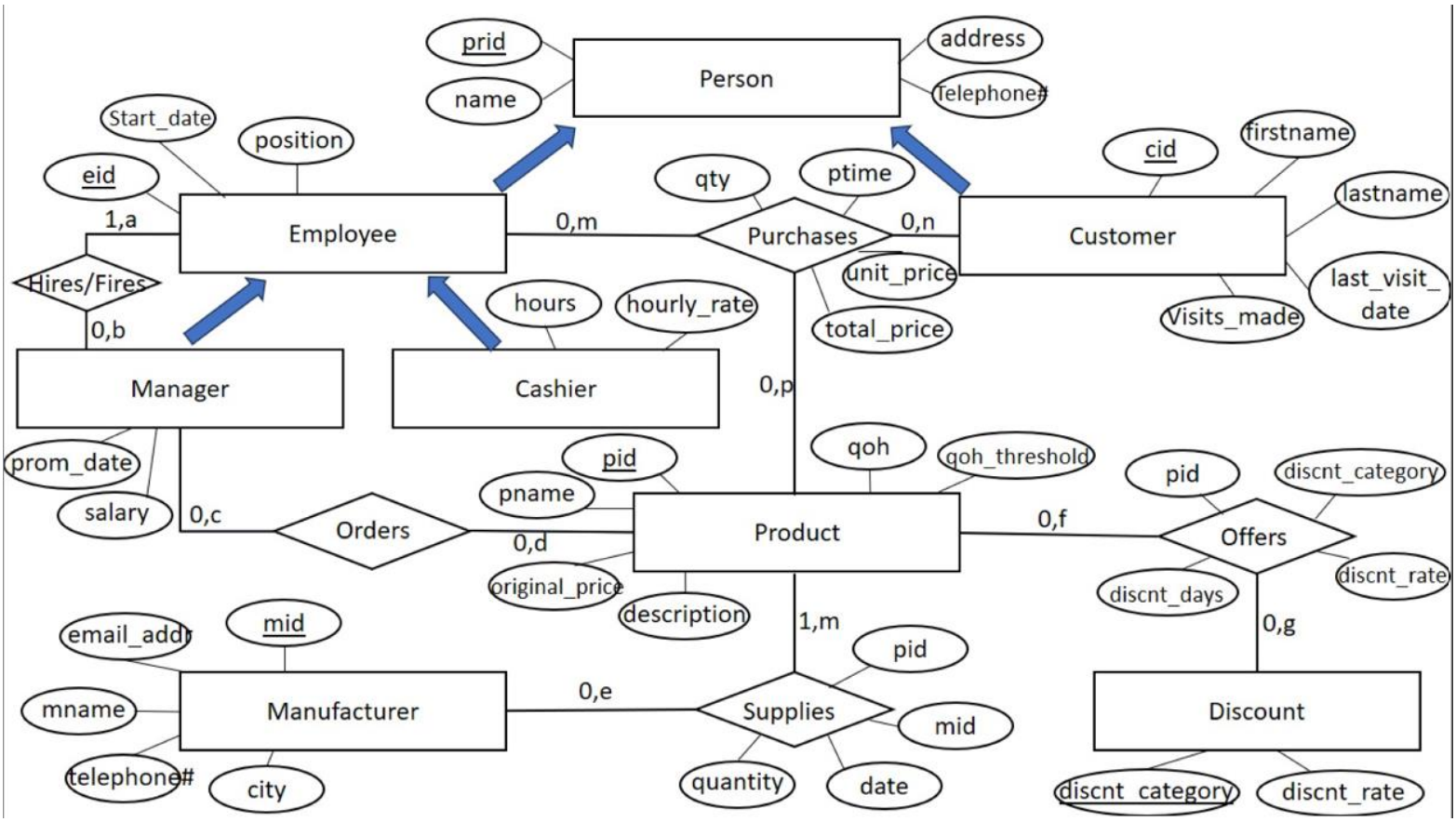
CS43/532 Homework 1

I understand that, if I get caught cheating (e.g., copying from any online/offline source) on this homework, I will receive a grade of zero for this work and my letter grade will be decreased by one full level. Also, I must fill in and sign the official form of the Watson College.

Name: Saurabh Patidar

Signature:_  _____ _____

Answer 1:



Answer 2:

A sub entity set should be used if its entities have significant special features (attributes and relationships) that are not shared by other entities in the original entity set. It can be a good idea to create a sub entity sets Cashiers and Managers for Employees because Cashiers and Managers both have special attributes. A Cashier has attributes 'hourly_rate' and 'hours' which are special for Cashier. A Manager has attributes 'prom_date' and 'salary' which are special for Manager. Also, managers do not have hourly rate and hours. In addition to that a Manager hires/fires employee. So, Manager can have a special relationship set 'hires/fires'.

If we do not create a sub entity sets Cashiers and Managers for Employees then we need to store the special data of both Managers and Cashiers in the Person entity which will cause lots of redundancy.

But again, this choice depends upon the database designer and the estimated number of Managers and Cashiers as well as the type of query results or views expected by application. To elaborate this further, it is mentioned that Managers can check sales statistics, order products, hire/fire employees, etc and Cashiers of the business to check product inventory and product sales discount information. Both Managers and Cashiers have different relations and expectations from the other data. To achieve all this, it can be a good idea to create a sub entity sets Cashiers and Managers for Employees.

Answer 3:

A super entity set should be used if it can significantly simplify the ER diagram, i.e., when the sub entity sets share significant common properties (attributes and relationships). In this case Employees and Clients have some common attributes like name, address, telephone#. Employees and Clients does not have any relationships in common. So, as per the current given requirement, super entity set can be created for them but this won't significantly simplify the ER diagram because of no common relationships and a few common attributes. So, it further depends on the use of these common attributes.

If multiple action needs to be performed on these common attributes and those actions are similar for both Employees and Clients/ Customers then creating super entity set for them can be a good idea. These actions can the related to retrieval, storage, business validations on the attribute value etc.

And since it is a good practice to support such scalability, would prefer to create the super entity set for Employees and Clients.

Answer 4:

There are some constraints in the requirements document that cannot be expressed using the ER diagram discussed in class. Please find below all such constraints:

- An employee can be either a manager or a cashier that is, the position has only two possible values, “cashier” and “manager”. There is no way to define the domain of the attribute ‘position ’ in the ‘Employee’ entity. But still ER diagram specifies what entities can inherit ‘Employee’ entity.
- Managers do not have hourly rate and hours.
- Cashiers who start to work on the same day have the same hourly rate. Cannot express in ER diagram ‘start_date’ → ‘hourly_rate’ for Cashier.
- Managers who are promoted at the same time have the same annual salary rate. Cannot express in ER diagram ‘prom_date’ → ‘salary’ for Managers.
- No cashier can work for more than 100 hours per pay period (two weeks).
- Different addresses have different telephone numbers. Cannot express in ER diagram ‘address’ → ‘telephone#’.
- Threshold that will trigger a reminder to increase inventory if the quantity on hand falls below the threshold (qoh_threshold). Cannot express the trigger in ER digram.
- Foreign keys: There are multiple foreign keys required for the mentioned design. Example: in relationship sets. But none of them can be expressed in ER diagrams discussed in class.
- A telephone number is required but email address is optional. Required and optional field constraints for this fields cannot be expressed in ER diagram.
- Different manufacturers have different telephone numbers and different email addresses (if exist).
- We further assume that the same manufacturer may supply the same product multiple times but no more than once each day. This could have been expressed by making composite primary key using pid, mid and date in the ‘Supplies’ relationship set.
- No customer can buy more quantity of a product than the quantity on hand of the product.
- All discounts that start on the same day with the same discount category will end on the same day later. Cannot express in ER diagram (‘discnt_category’, ‘discnt_start_date’) → ‘discnt_days’ for ‘Offers’ relationship set.