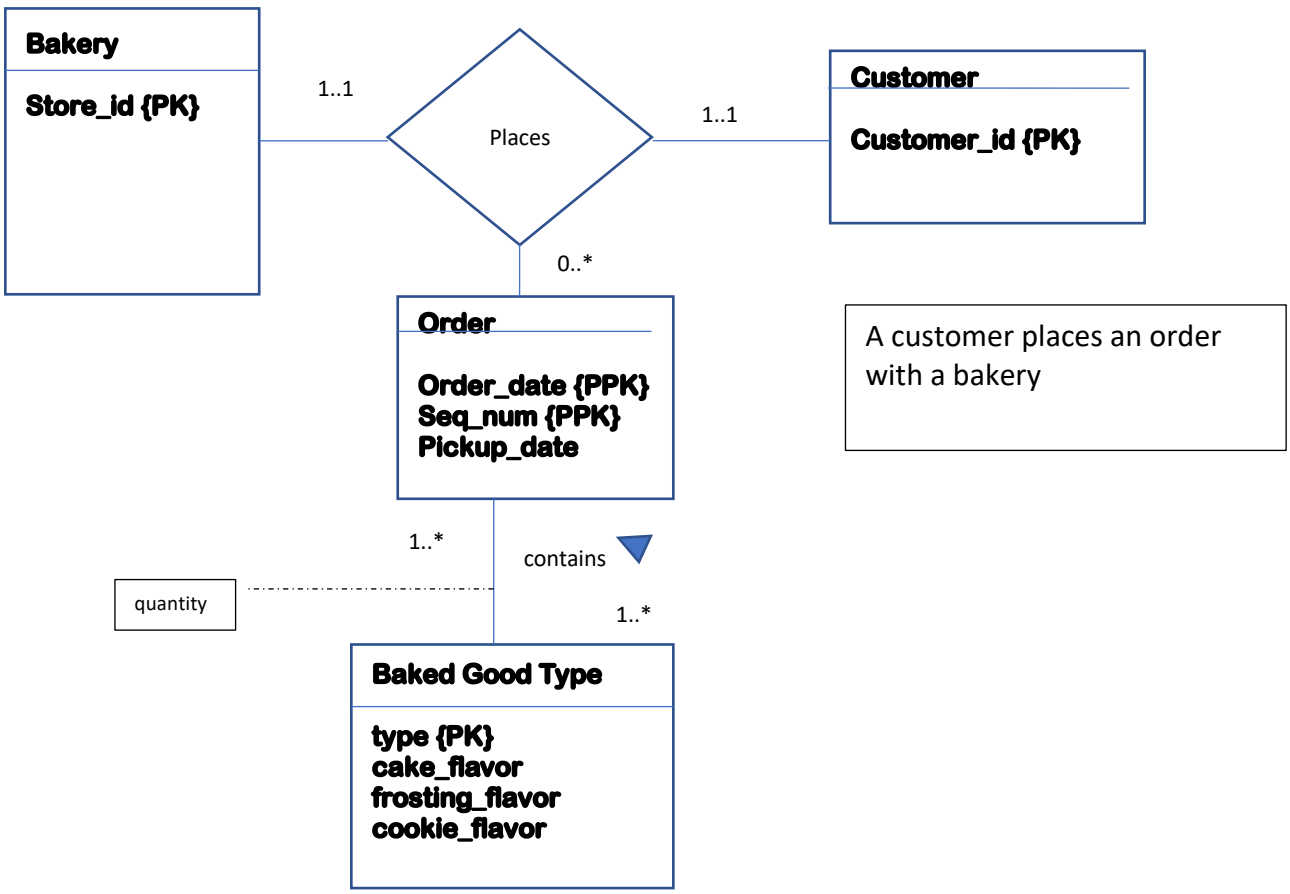
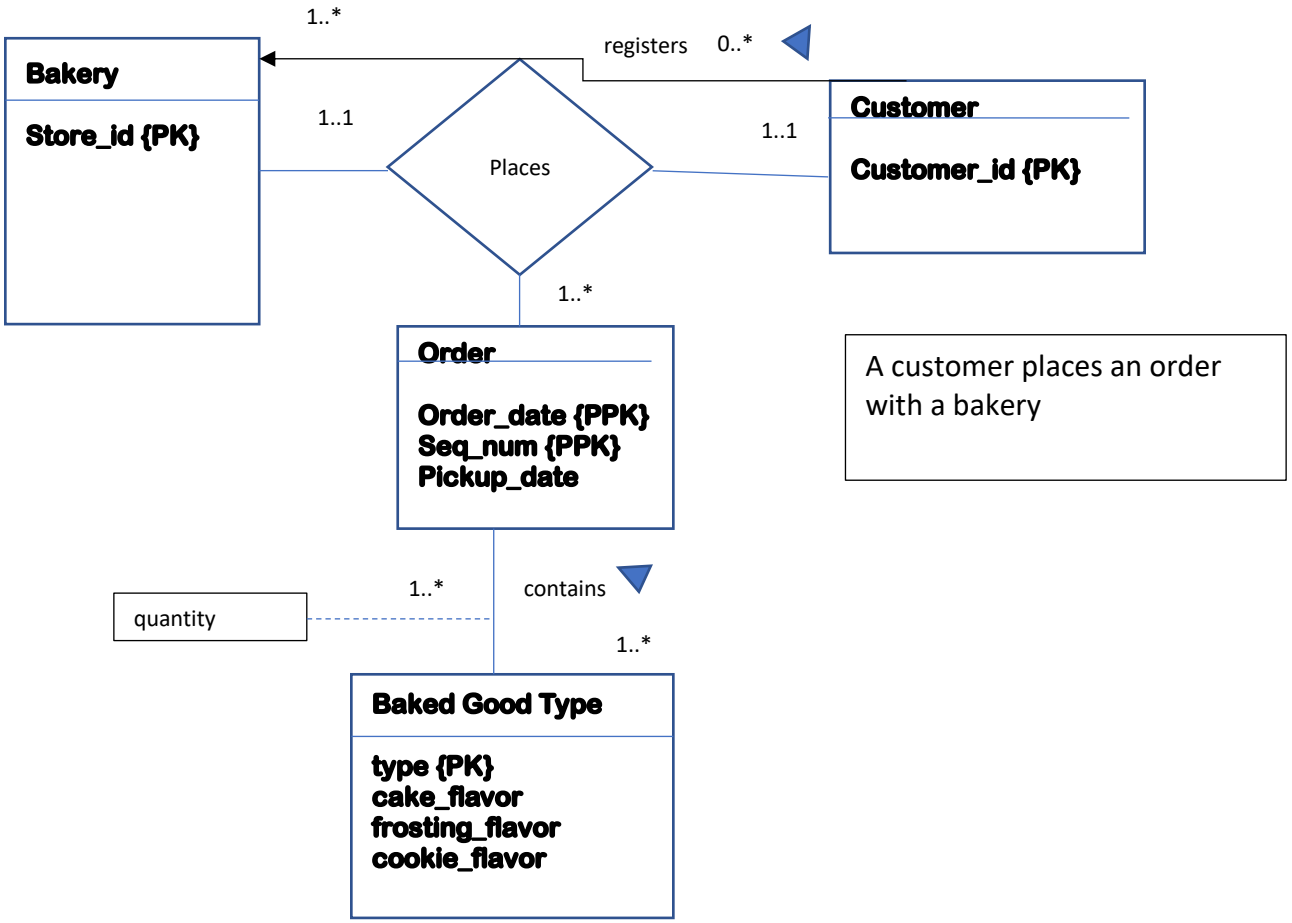
1

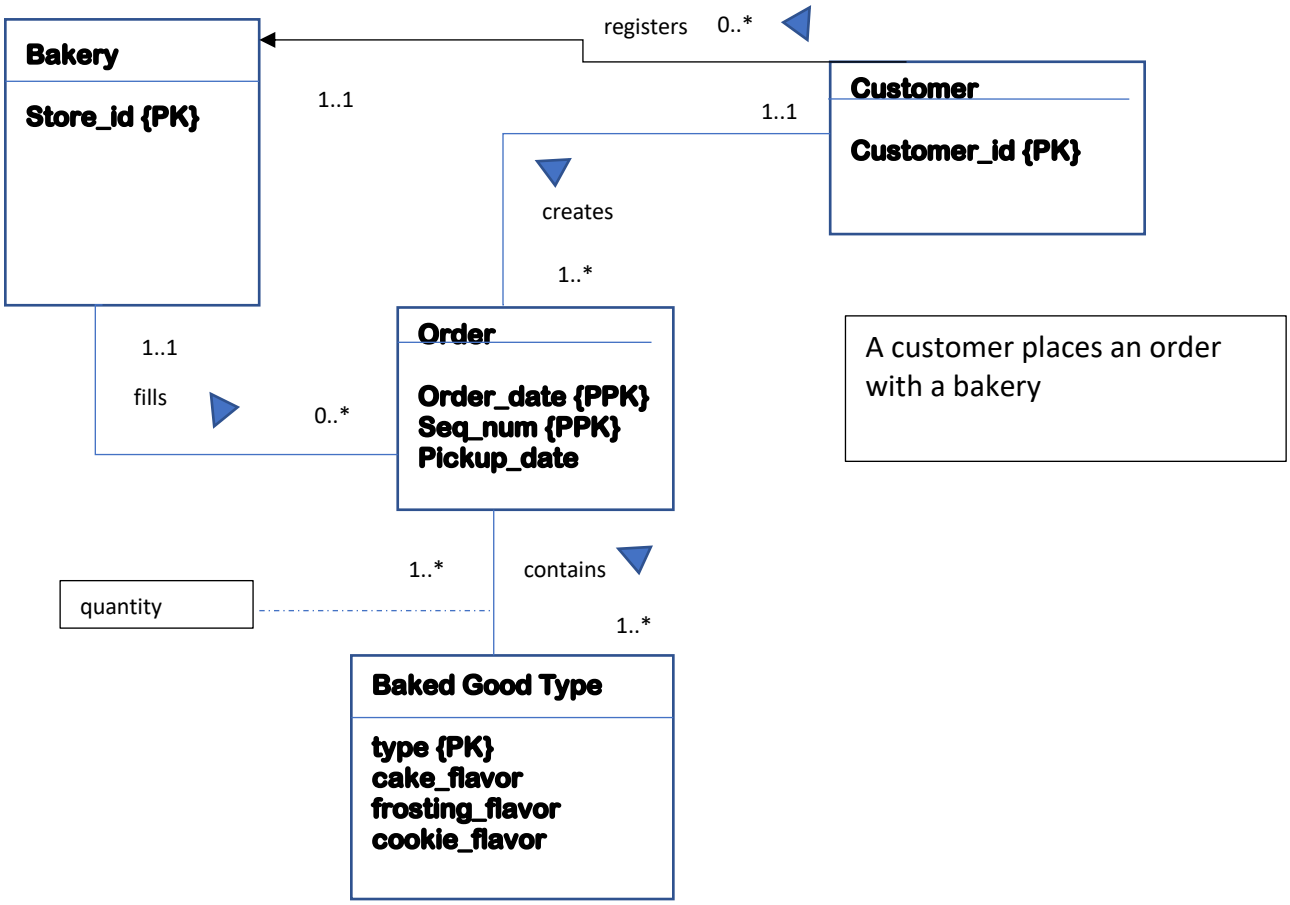
Below is the simplest representation but the least informative. We are not representing the different processes where a person becomes a customer. We are also representing one entity for all the different baked good types.



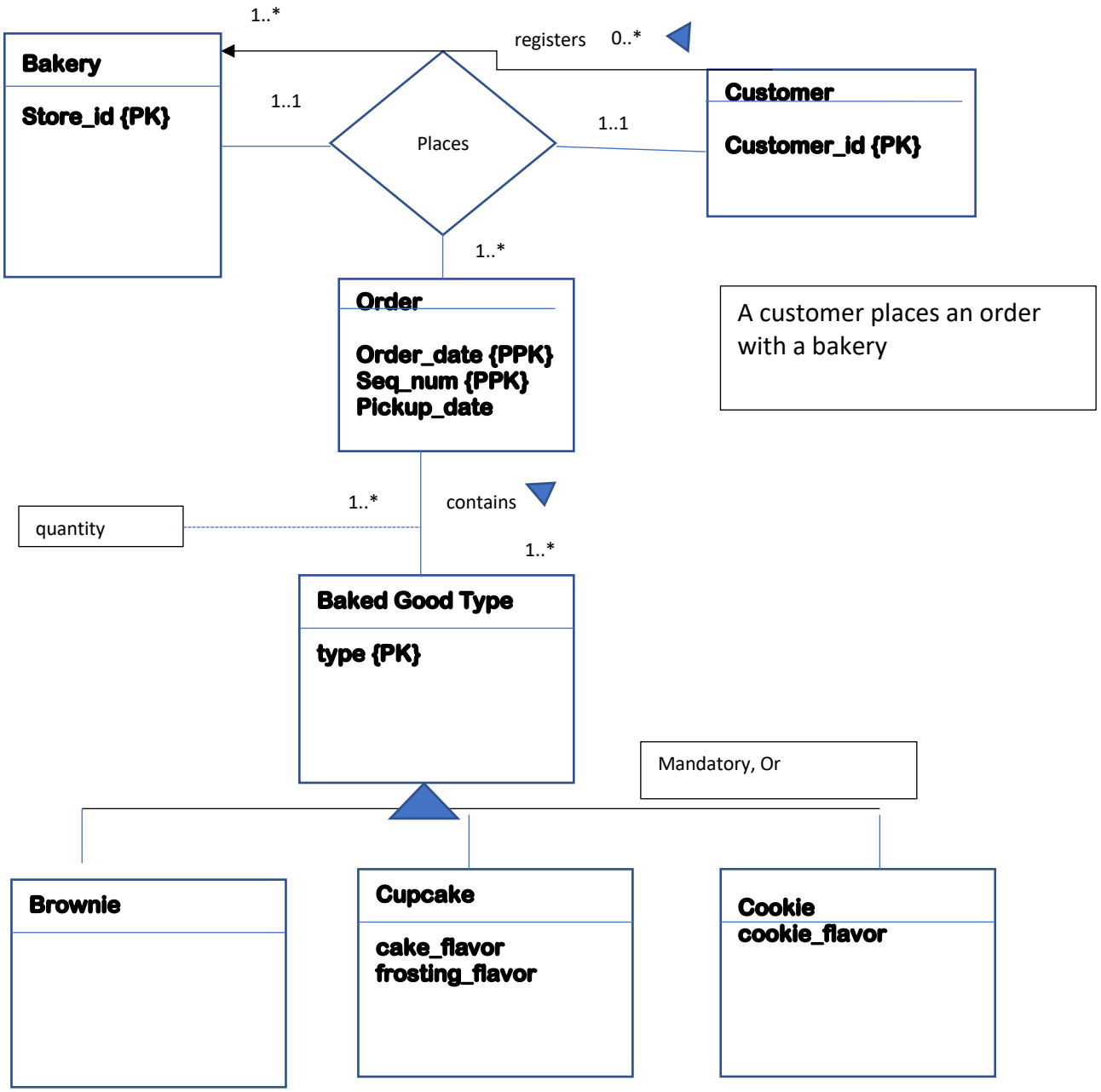
We can add 2 different relationships to represent the 2 different methods where a person can become a customer.



We can also choose to not use a ternary relationship for the places order relationship. We can break it up into 2 binary relationships. A customer creates an order and an order is filled by a bakery.



We can also choose to use the superclass/subclass relationship to be more specific with the attributes associated with the entity baked good type.



If you leave off the subclass brownie, because it has no attributes, the participation constraint should be {Optional, Or}, since we are not representing every baked good type.

We can choose to represent customers as a superclass/subclass, this way we can represent the subclasses' relationships more precisely.

We can also choose to use the superclass/subclass relationship to be more specific with the attributes associated with the entity baked good type.