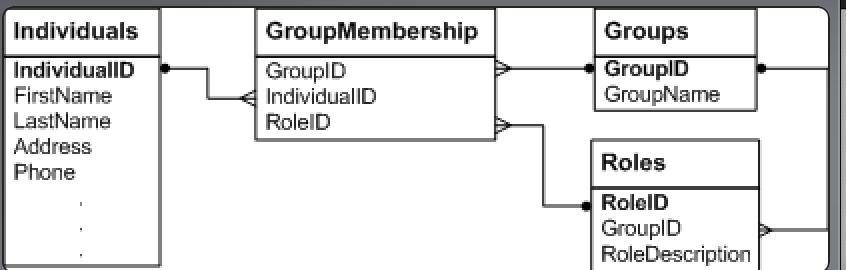
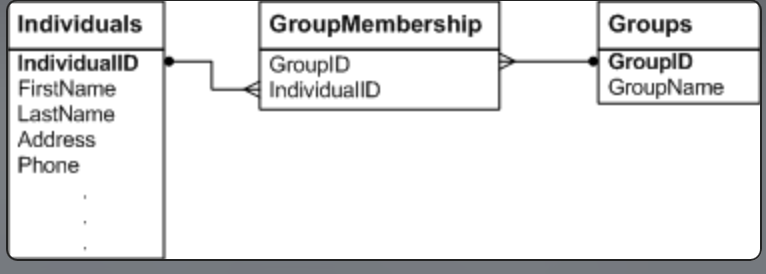
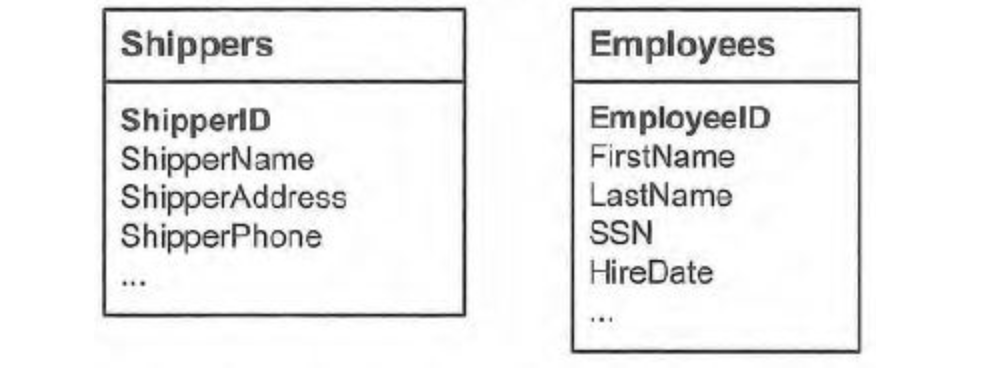


A customer could have many orders and their id would be the foreign key that would relate them to the orders. An order could have multiple line items and the order id would relate the order to the the line items. A product could occur in many line items and the product id would be unique to the product, but could be in many line items as the foreign key

Shipper ID



I’m having trouble getting the formatting to work, but I’d add shipper id and employee is to the orders table, where they would be foreign keys to the primary keys in the shippers and employees tables respectively. One shipper could ship many orders and one employee could have many orders. So, each of these tables would have a one to many relationship with the orders table.

3. I would think the three foreign keys in the orders table (customer ID, shipper ID and employee ID) should all be indexed as they are foreign keys, will have a large volume of distinct values, would be updated rarely since the IDs are unlikely to change, and would be useful in repeated searches. The order id and product id in the Order line items table should also be indexed for the same reasons.

4. This answer from the exercises folder shows that each group membership could have many individuals and many groups, each distinguished by their respective IDs. In this way, there would be a many to many relationship.

5. Here we can see that each group can have many roles while membership can only have one specific role.