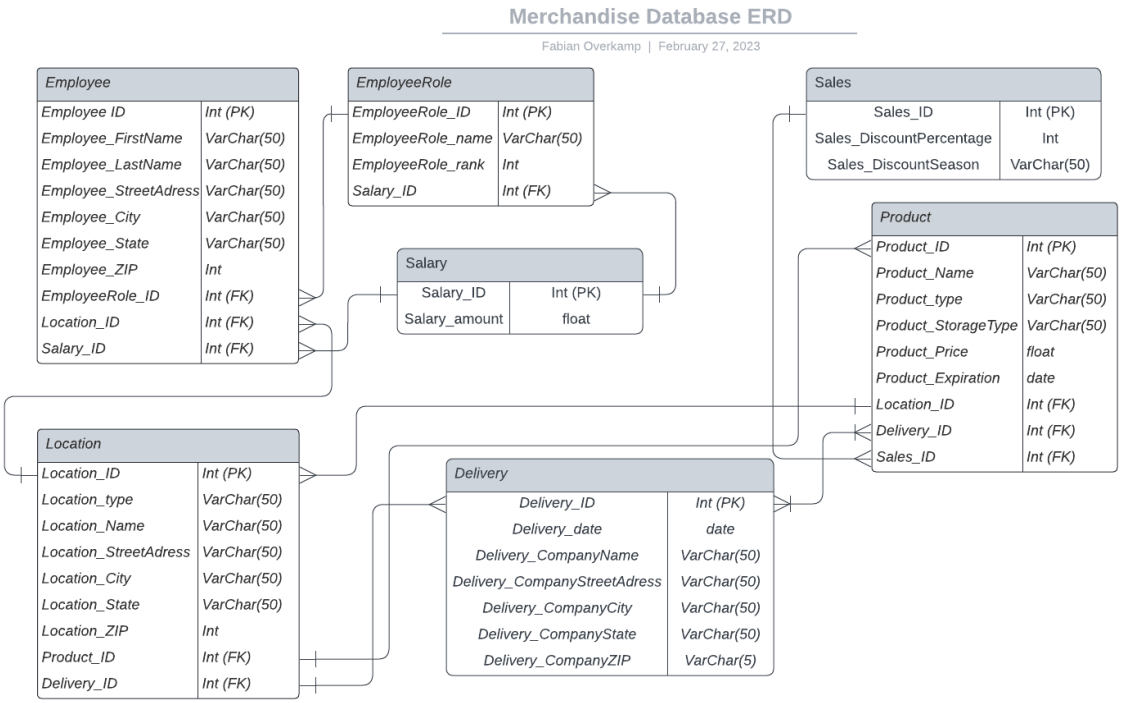

My Final Project is about building a useful and effective Database for a merchandise. It should be able to provide all the important information for the boss, the employee, and the customers.

One of the tables will be the product table. The product table includes all the products available and their details. The table includes a foreign key to the location table, the delivery table, and the sales table as they all provide further important information about the products. Usually, this table would include Millions of rows as a merchandise usually sells a huge amount of products and even if products are exactly the same, the ID will distinguish them so that every single product uses one row. I probably won't have the time to enter so much data, which is why I will limit myself to a small variety and amount of products. I guess I am gonna have around 50 rows.

Another table will be the sales table, which includes sales for special events. It is independent and only connects to the product table. The sales table will have around 5 - 10 rows estimated.

Next is the Delivery table. The delivery tale gives all the important information about where a delivery comes from, who sent it, and when and where it arrived. In this case, I am a little confused because the product table needs the delivery table and vice versa (same thing for "delivery" and "location"). I am not sure if I need a foreign key for both tables or if one foreign key is enough to connect the tables in both directions. In the real world, the delivery table would

again use a huge amount of rows for every delivery that was made but for my project, I will probably enter up to 5 and I won't be updating it as it would be happening in real life.

I will also have a location table that includes all the important information about the different shops and storage locations. It's dependent on the product and the delivery. The number of locations is usually not too big so I will have around 5 locations.

Next, we got the Employee table which has everything to know about the workers. The table is dependent on EmployeeRole, location, and Salary. I will have around 20 employees, so 20 rows.

The EmployeeRole table lists all possible jobs and the ranks that you can achieve inside a role. It gives the Employee table his data and gets data from the salary table. There will be around 3 different job types.

The salary table gives you the amount of money the workers earn. It shares the data with the employee table and the EmployeeRole table. The number of rows depend if the number of possible roles and ranks are higher or the number of employees. In my case, the number of roles and ranks will be higher so I will have around 30 salary rows.